UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Cícero Augusto Magalhães da Silva Neves

Um Método para Construir Intervalos de Predição Sensível ao Ruído em Redes Neurais

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

Prof. Mauro Roisenberg, Dr.
Orientador

Florianópolis, Setembro de 2009

Um Método para Construir Intervalos de Predição Sensível ao Ruído em Redes Neurais

Cícero Augusto Magalhães da Silva Neves

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Redes Neurais e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

_____

Prof. Mauro Roisenberg, Dr.

Coordenador do Curso

Banca Examinadora

_____

Prof. Mauro Roisenberg, Dr.

Orientador

_____

_____

Profa. Jacqueline Gisèle Rolim, Dra.

_____

Guenther Schwedersky Neto, Dr.

_____

Prof. Dalton Francisco de Andrade, PhD.

“You’ve achieved success in your field when you don’t know whether what you’re doing is work or play." (Warren Beatty)
“Nada existe de permanente a não ser a mudança." (Herácito)

Agradecimentos

Gostaria de dedicar este espaço da minha dissertação para expressar meus agradecimentos a todas as pessoas sem as quais eu não conseguiria chegar até onde cheguei. A dissertação de mestrado, apesar de envolver a princípio somente uma pessoa, o mestrando, exige inúmeros esforços dos indivíduos que a ele estão relacionados, quer seja por laços afetivos ou profissionais.

Em primeiro lugar gostaria de agradecer aos meus pais por todo o amor e por todos os sacrifícios feitos em função dos meus estudos. A educação e lições passadas por eles me prepararam para as inúmeras dificuldades que enfrentei durante todo o período do mestrado.

Não posso deixar de citar também minha irmã e amiga, Cibele, com a qual sempre pude contar para me ouvir e ajudar a resolver meus problemas mais difíceis.

Faço um agradecimento muito especial ao amor da minha vida, Bruna, que foi a pessoa mais próxima de mim durante todo o processo de pesquisa e escrita desta dissertação, sendo amiga, companheira, compreensiva, e tendo sempre uma palavra de conforto nas ocasiões em que eu me via perdido. O apoio e carinho dela foram extremamente importantes para que eu conseguisse concluir este trabalho.

Agradeço também ao Dino, Lia e Rafa por toda a preocupação e carinho despendidos a mim. Estas pessoas me acolheram todas as vezes em que precisei de um lar e estava impossibilitado de ir para casa. Gostaria que eles soubessem que os considero como uma segunda família e serei eternamente grato a eles.

Ao Professor Mauro, agradeço pela orientação e paciência com minhas falhas durante todo o período do mestrado. Seus ensinamentos foram responsáveis pela

minha evolução profissional e por aumentar o meu gosto pela pesquisa científica.

Por fim agradeço à Petrobras, particularmente ao CENPES e aos seus funcionários, pelos investimentos técnico e financeiro que possibilitaram a execução do projeto que serviu de inspiração para esta dissertação.

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

Devido a sua grande capacidade representacional, as Redes Neurais Artificiais (RNAs) têm sido largamente utilizadas como aproximadores universais de funções na construção de modelos preditivos não-lineares, inclusive em aplicações da indústria petrolífera. Contudo, devido à natureza estocástica do treinamento das RNAs, indicadores de qualidade e confiabilidade para as saídas destes modelos, como os intervalos de predição, são extremamente necessários e desempenham um papel importante em aplicações reais.

Muitas das técnicas adotadas para o cálculo dos intervalos de predição estabelecem uma série de restrições que devem ser atendidas pelos dados amostrais usados para treinar a RNA. Uma dessas restrições impõe que a variância dos resíduos seja constante, fato que nem sempre ocorre em aplicações reais, onde a taxa de ruído existente pode variar como função dos dados de entrada, fazendo com que a confiabilidade calculada pelos métodos tradicionais não seja condizente com a real precisão da rede neural.

Nesta dissertação, uma extensão para um método de cálculo de intervalos de predição para redes neurais baseado na teoria da regressão não-linear é apresentada. A idéia principal do método proposto consiste em utilizar técnicas de agrupamento para estimar a variância dos resíduos em função do vetor de entrada apresentado à rede e incorporar essa estimativa ao cálculo dos intervalos de predição. Os resultados dos experimentos realizados mostram que tal abordagem pode gerar intervalos de predição com uma melhor representação da precisão das respostas das RNAs.

Palavras-chave: redes neurais, confiabilidade, intervalos de predição, agrupamento.

Abstract

Due to their large representational capacity, Artificial Neural Networks (ANNs) have been widely used as universal approximators of functions in the construction of nonlinear predictive models, including oil industry applications. However, due to the stochastic nature of the ANN training, indicators of quality and reliability to the outputs of these models, such as the prediction intervals, are extremely necessary and play an important role in real applications.

Many of the techniques adopted to calculate the prediction intervals impose a set of constraints that must be respected by the data sample used to train the ANN. One of these restrictions requires that the variance of the residuals must be constant, a fact that does not always occur in real applications, where the existent noise rate may vary as a function of input data, making the reliability calculated by traditional methods become not suitable with the actual accuracy of the neural network.

In this work, an extension to a method for calculating prediction intervals for neural networks, based on the theory of nonlinear regression, is presented. The main idea of the proposed method is to estimate an input dependent variance of the residuals using clustering techniques and incorporate this estimate to the computation of the prediction intervals. The results of the performed experiments show that this approach can lead to prediction intervals which better represent the accuracy of the ANNs outputs.

Keywords: neural networks, reliability, prediction intervals, clustering.

Capítulo 1

Introdução

1.1 Contextualização

Redes Neurais Artificiais (RNAs) são ferramentas computacionais bastante utilizadas para tarefas de regressão e classificação em aplicações nas áreas da medicina, indústria e mercado financeiro [Haykin 1998]. Uma das características mais marcantes destas ferramentas é a sua capacidade de aprendizagem a partir de exemplos, o que as torna forte candidatas quando se deseja modelar um sistema onde o relacionamento entre as variáveis que o compõem é por demasiado complexo.

Particularmente, em problemas relacionados à caracterização de reservatórios de petróleo, as RNAs têm sido consideradas como uma ferramenta de auxílio poderosa para os geoestatísticos em tarefas de predição de propriedades petrofísicas de reservatórios a partir de informações retiradas de poços perfurados e dados sísmicos [Wong et al. 2002], [Yang et al. 2002].

Apesar dos bons resultados normalmente obtidos pelas redes neurais em aplicações como esta, o desempenho das mesmas está constantemente atrelado tanto a fatores de natureza estrutural (quantidade de neurônios, quantidade de camadas, etc.) como a fatores relacionados aos dados disponíveis durante o processo de treinamento (quantidade de ruído nos dados, representatividade do sistema a ser modelado por parte do conjunto de treinamento, etc.). Estes fatores, juntamente com a natureza estocástica

do treinamento das redes neurais, fazem com que estes modelos matemáticos careçam de indicadores que retratem a confiabilidade das respostas fornecidas por eles [Papadopoulos et al. 2001], [Chryssolouris et al. 1996], [Leonard et al. 1992].

Tradicionalmente, um indicador usado para medir o desempenho de uma RNA é o *Erro Médio Quadrático* ou EMQ, o qual é representado pela soma das diferenças ao quadrado entre as respostas da RNA e os respectivos valores reais (ou alvos) disponíveis no conjunto de treinamento, dividida pela quantidade de exemplos contidas neste conjunto. Mesmo sendo frequentemente utilizado, inclusive pelos algoritmos de treinamento, como um meio de avaliar quão boa é a representação do sistema de interesse pela rede, o EMQ serve apenas como um indicador global de desempenho não sendo adequado para se estimar a confiabilidade de uma única resposta da rede neural [Leonard et al. 1992].

A obtenção de um indicador local de confiabilidade para as respostas de RNAs tem sido objeto de estudo de muitas pesquisas, sendo o cálculo de *Intervalos de Predição* (IPs) a abordagem mais comumente adotada nos trabalhos já realizados [Chryssolouris et al. 1996], [Leonard et al. 1992], [Nix and Weigend 1995], [Shao et al. 1997], [Wong et al. 2002], [Chinman and Ding 1998]. Um IP delimita uma região dentro da qual se espera que o verdadeiro valor da variável representada pela saída da rede esteja localizado. A probabilidade deste valor realmente se encontrar dentro do intervalo é determinada pelo nível de confiança para o qual o IP foi calculado, de forma que quanto maior for o intervalo de predição obtido, maior deverá ser a incerteza atribuída à estimativa da rede para este valor.

1.2 Motivação

De forma geral os métodos utilizados para se estimar a confiança das respostas de uma RNA através de intervalos de predição podem ser divididos em dois grandes grupos. O primeiro grupo possui um caráter mais global, utilizando informações que representam características dos dados de treinamento como um todo. Dentro deste grupo estão as abordagens calcadas na teoria da regressão não-linear, como as propostas

feitas em [Chryssolouris et al. 1996] e [Veaux et al. 1998]; as abordagens que fazem uso de técnicas de *bootstrapping* [Lebaron and Weigend 1994] e [Heskes 1997]; e por fim, métodos apoiados em lógica *fuzzy* [Wong et al. 2002].

Técnicas de *bootstrapping*, apesar de apresentarem bons resultados em estudos comparativos, exigem um grande esforço computacional [Dybowski and Roberts 2001], [Papadopoulos et al. 2001]. Por outro lado, o método proposto em [Wong et al. 2002], apesar dos baixos custos de implementação e execução, carece de um melhor embasamento estatístico e é em certos aspectos muito subjetivo. Finalmente, o método proposto em [Chryssolouris et al. 1996], modificado por *De Veaux* em [Veaux et al. 1998] para se ajustar a casos em que o conjunto de treinamento é pequeno, é relativamente simples de ser implementado e obteve bons resultados em trabalhos como [Papadopoulos et al. 2001] e [Yang et al. 2002]. Tal método é derivado da teoria de regressão não-linear vista em [Seber and Wild 2003] e possui custo computacional e estabilidade melhores que as técnicas de *bootstrapping* [Dybowski and Roberts 2001].

A abordagem adotada em [Chryssolouris et al. 1996], contudo, exige que certas hipóteses sejam tomadas como verdadeiras. É preciso que a rede neural seja treinada até a convergência para um ponto de mínimo na superfície de erro; que não haja erros de medição das variáveis de entrada; que os resíduos sejam identicamente e independentemente distribuídos (i.i.d.); e que tais resíduos possuam uma distribuição na forma $N(0, \sigma^2)$, i.e., distribuídos normalmente com média zero e variância constante [Chryssolouris et al. 1996]. Estas suposições limitam o uso da abordagem em questão uma vez que as mesmas raramente são satisfeitas na prática [Yang et al. 2002].

Visando contornar estas limitações, o segundo grupo de métodos tenta extrair atributos mais individualizados (específicos para cada dado de entrada), apresentando um modo de operar mais local. Exemplos deste tipo de abordagem podem ser encontrados em [Leonard et al. 1992], [Shao et al. 1997], [Chinman and Ding 1998] e [Nix and Weigend 1995]. Contudo, estas técnicas estão limitadas a determinado tipo de arquitetura de redes neurais [Leonard et al. 1992], ou tratam o modelo de rede neural utilizado de forma não muito bem fundamentada [Chinman and Ding 1998], ou ainda, exigem processos e estruturas adicionais que acabam por aumentar a complexidade do

processo como um todo [Nix and Weigend 1995], [Shao et al. 1997].

A necessidade de indicadores de confiança consistentes para as respostas de redes neurais em aplicações reais e a falta de métodos de baixo custo computacional e ao mesmo tempo embasados na teoria de modelos de regressão não-linear fornecem uma ótima motivação para que pesquisas sejam feitas nesta área. O presente trabalho propõe uma extensão para um método já existente a fim de que o mesmo possa ser aplicado em problemas onde a variância dos resíduos não é constante.

1.3 Objetivos Geral e Específicos

1.3.1 Objetivo Geral

O objetivo desta dissertação é apresentar uma extensão ao método apresentado em [Chryssolouris et al. 1996], capaz de calcular a exatidão das respostas de uma RNA, ao mesmo tempo em que modela a relação de dependência entre a distribuição do ruído e os dados de entrada da rede neural.

1.3.2 Objetivos específicos

Para alcançar o objetivo principal deste trabalho os seguintes objetivos específicos devem ser alcançados:

1. Realizar uma revisão crítica e detalhada dos trabalhos existentes na área na qual está o foco deste estudo.

2. Estudar de que maneira a distribuição do ruído pode ser estimada localmente.

3. Buscar um modo coerente e estatisticamente correto de se modelar e executar experimentos que ilustrem as vantagens do método aqui proposto sobre método apresentado em [Chryssolouris et al. 1996].

4. Realizar os experimentos e colher as informações necessárias para se comparar o desempenho dos dois métodos.

1.4 Estrutura da dissertação

Esta dissertação está organizada da seguinte maneira:

O Capítulo 2 o conceito de modelos de regressão não-linear é introduzido junto com a teoria existente para o cálculo de intervalos de predição. Além disso, uma analogia entre estes modelos e redes neurais MLP, explicando-se como tal teoria pode ser aplicada nestas ferramentas, e uma revisão bibliográfica dos métodos normalmente aplicados são feitas.

No Capítulo 3 as consequências de se utilizar o método proposto em [Chryssolouris et al. 1996], quando a variância dos resíduos não é constante, são constatadas e a proposta deste trabalho é apresentada.

O Capítulo 4 contém a descrição dos experimentos realizados, tanto para problemas artificialmente criados como para problemas de caracterização de reservatórios, e discussão dos resultados obtidos.

Por fim, no Capítulo 5 são apresentadas as conclusões, retomando os principais aspectos abordados e apresentando sugestões para trabalhos futuros.

Capítulo 2

Estimativa de confiança para regressão não-linear e RNAs

Neste capítulo os modelos de regressão não-linear e a teoria que possibilita o cálculo intervalos de predição através destes modelos são descritos. A partir do delineamento destes conceitos, discorre-se sobre a aplicação dos mesmos quando do uso de redes neurais, e por fim, é apresentada uma revisão crítica dos principais trabalhos e propostas encontradas na literatura relacionada ao tema, discutindo-se vantagens, desvantagens e restrições das mesmas, visando justificar a escolha do método selecionado como foco no presente trabalho.

2.1 Regressão não-linear

Um modelo de regressão não-linear é uma ferramenta estatística usada para estabelecer um relacionamento entre um conjunto de variáveis onde pelo menos uma dessas variáveis é aleatória. Neste conjunto uma das variáveis é chamada de variável de interesse ou variável dependente e é representada por y. As demais variáveis, denotadas comumente por $x_1, x_2, x_3, ..., x_k$ são denominadas de variáveis independentes e são usadas na tentativa de predizer o comportamento de y. Daqui em diante o conjunto, formado por uma observação de cada variável independente, será referenciado pelo vetor $\mathbf{x}$.

Com o apoio de observações empíricas ou teorias que indiquem algum tipo de relacionamento entre a variável de interesse e as variáveis independentes, tentativas são feitas a fim de encontrar um modelo que melhor se aproxime do tipo de relação existente entre as mesmas. Em outras palavras assume-se que a relação investigada pertence a uma família paramétrica de funções onde todos os elementos são conhecidos com exceção de um vetor de parâmetros $\theta = (\theta_1, \theta_2, ..., \theta_p)$ que precisa ser estimado [Seber and Wild 2003]. Além disso, estes parâmetros devem necessariamente estar combinados de uma maneira não-linear na função que representa o modelo escolhido. Desta forma, o modelo utilizado para a regressão pode ser escrito como:

$$y \approx f(\mathbf{x}_i; \theta^*) \tag{2.1}$$

onde $\mathbf{x}_i$ representa o i-ésimo vetor de observações de variáveis independentes e θ^* denota os valores ideais do vetor de parâmetros para a função que modela o sistema de interesse.

Mesmo que o modelo obtido se adeque razoavelmente nas observações usadas para estimar o vetor de parâmetros, devido a flutuações aleatórias ou erros de medição das variáveis ele provavelmente não irá fornecer uma resposta correta para todas as entradas que lhe forem apresentadas, dando origem a discrepâncias entre as respostas do modelo e os valores reais [Seber and Wild 2003]. À diferença entre os resultados obtidos pelo modelo e as medições realizadas dá-se o nome de *erro residual* ou *resíduo,* e a mesma é representada pela notação ε_i, onde i indica a observação usada para obter este erro. Desta forma o sistema que tenta-se modelar, a partir de n observações, é representado como:

$$y_i = f(\mathbf{x}_i; \theta^*) + \varepsilon_i \quad i = 1, 2, ..., n \tag{2.2}$$

Para um melhor entendimento um exemplo será apresentado. Tem-se à disposição um conjunto de 58 observações, compiladas pelo *World Watch Institute* [Boggs 2009], referentes à medição em partes por milhão (ppm) da concentração de dióxido de carbono na atmosfera do ano de 1764 ao ano de 1995. A variável referente ao ano de medição será denotada por x enquanto que a concentração de dióxido de carbono será

representada por y. Plotando essas 58 observações em um gráfico tem-se o resultado obtido na Figura 2.1.

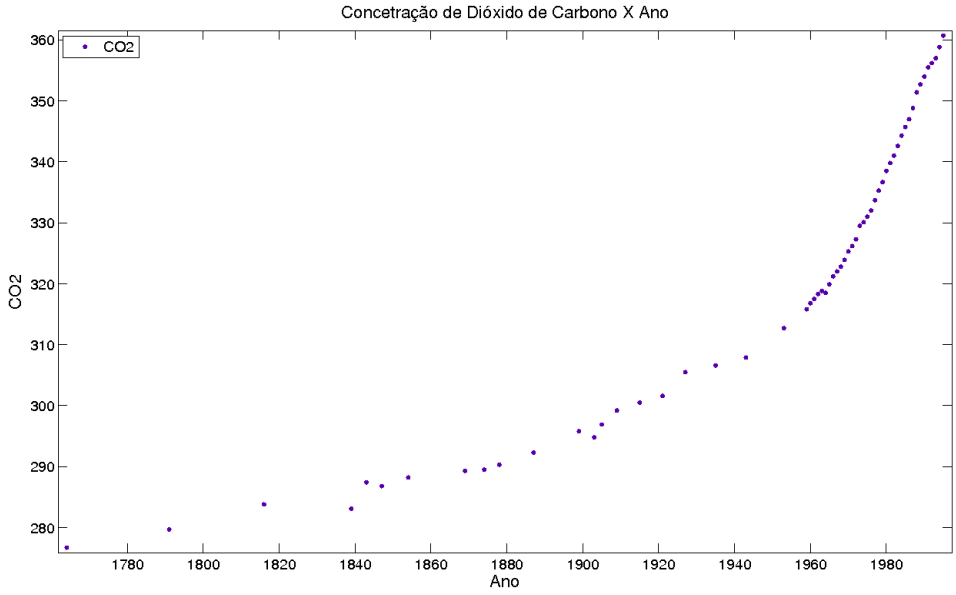

Figura 2.1: Gráfico das 58 observações da concentração de CO_2 em função dos anos

Observando esta figura pode-se perceber a existência de algum tipo de relação dentre as duas variáveis, o que leva à suposição de que uma determinada família de funções consiga representar satisfatoriamente essa relação. A seguinte família de funções foi escolhida:

$$f(x) = ae^{(bx)} + ce^{(dx)} \tag{2.3}$$

Onde o vetor de parâmetros, representado por $\theta = (a, b, c, d)$ foi estimado como $\hat{\theta} = (2,664 \times 10^{-12}, 0,01562, 452,6, -0,0002689)$. Construindo um gráfico da curva da função juntamente com as observações (Figura 2.2), pode-se assumir que a relação entre a concentração de CO_2 e o ano em que a medição foi feita está bem representada pelo modelo obtido.

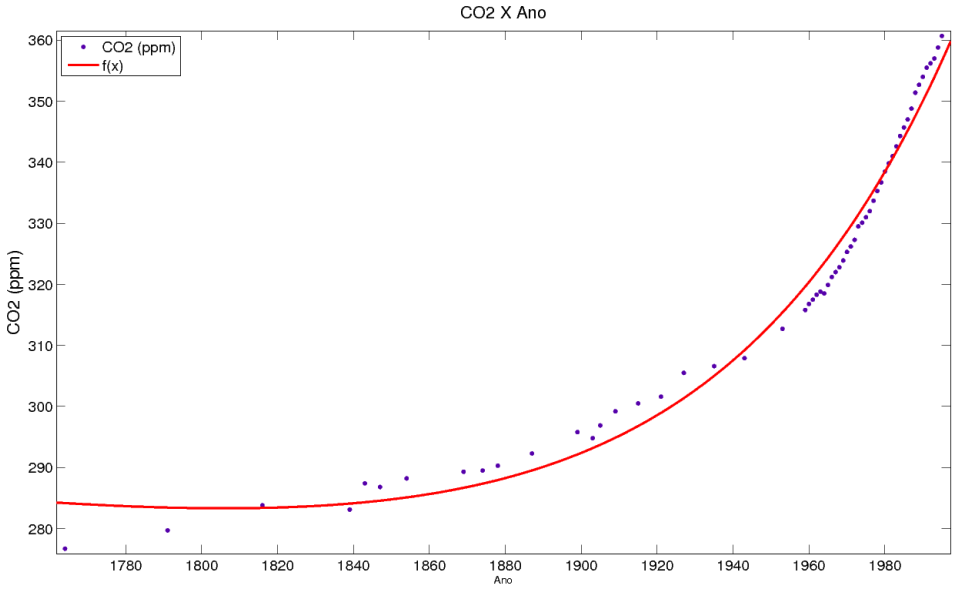

Figura 2.2: Gráfico do curva da função obtida juntamente com as 58 observações da concentração de CO_2 em função dos anos

Tomando, por exemplo, a observação do conjunto de dados referente ao ano de 1764, tem-se que a concentração de CO_2 medida neste ano foi de 276,6 ppm. Ao fornecermos este mesmo ano como entrada para a função temos como resposta uma concentração de 284,167 ppm. Esta diferença de 7,576 ppm entre o resultado da função e do valor medido é o resíduo mencionado anteriormente, que para este exemplo seria denotado por ε_1 uma vez que o mesmo se refere à primeira observação do conjunto de dados utilizado.

A soma dos quadrados dos resíduos é normalmente usada pelos métodos de otimização de parâmetros apresentados no Apêndice A, a fim de se obter o vetor de estimativas dos parâmetros $\hat{\theta}$. É lógico considerar, portanto, que os resíduos sirvam como indicadores de quão bem o modelo obtido correlaciona a variável de interesse e o conjunto de variáveis independentes. Esta soma, contudo, serve somente como indicador da qualidade global do modelo, não podendo ser usada para se estimar a qualidade de uma resposta específica do mesmo. Na próxima seção um indicador mais adequado para tal fim será apresentado.

2.2 Intervalos de Predição

Após um modelo de regressão não-linear ser obtido para um determinado sistema, o mesmo pode ser utilizado para diferentes fins. Dentre estes está o uso para a predição de um novo valor y_0 da variável de interesse a partir de um novo vetor de observações $\mathbf{x}_0$ independente dos demais usados para se obter o modelo de regressão [Seber and Wild 2003]. A resposta do modelo será uma estimativa $\hat{y}_0$ deste novo valor e sendo esta um valor estimado e não um valor real, uma medida de confiabilidade para $\hat{y}_0$ apresenta-se como uma informação valiosa.

Um Intervalo de Predição ou IP delimita uma região onde se espera que, com um determinado grau de confiança, uma nova observação da variável de interesse y_0 esteja localizada. A seguir, um dos métodos mais usados para obter esse indicador de confiabilidade, chamado de *método delta*, será apresentado tomando como referência [Seber and Wild 2003] e [Chryssolouris et al. 1996].

Para um sistema representado pela Equação (2.2), a soma dos erros quadráticos, usada para estimar o vetor de parâmetros $\hat{\theta}$ do modelo, é dada por:

$$S(\theta) = \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i; \theta))^2 \tag{2.4}$$

e portanto a estimativa do modelo obtido para uma nova observação $\mathbf{x}_0$ é obtida através de:

$$\hat{y}_0 = f(\mathbf{x}_0; \hat{\theta}) \tag{2.5}$$

Considerando que um valor suficientemente grande de n foi usado para se obter o vetor $\hat{\theta}$, é válido assumir que $\hat{\theta}$ esteja próximo do vetor de valores reais dos parâmetros θ^* permitindo desta forma que uma expansão em série de Taylor de primeira ordem seja usada para aproximar $f(x_0; \hat{\theta})$ em termos de $f(x_0; \theta^*)$ [Seber and Wild 2003]:

$$f(\mathbf{x}_0; \hat{\theta}) \approx f(\mathbf{x}_0; \theta^*) + \mathbf{f}_0^{\mathbf{T}} (\hat{\theta} - \theta^*) \tag{2.6}$$

onde

$$\mathbf{f_0^T} = \left(\frac{\partial f(\mathbf{x}_0; \theta^*)}{\partial \theta_1^*}, \frac{\partial f(\mathbf{x}_0; \theta^*)}{\partial \theta_2^*}, \cdots, \frac{\partial f(\mathbf{x}_0; \theta^*)}{\partial \theta_p^*} \right)$$

Calculando a diferença entre o valor real y_0 e a predição feita pelo modelo $\hat{y}_0$, usando as Equações (2.2) e (2.6), obtém-se:

$$y_0 - \hat{y}_0 \approx y_0 - f(\mathbf{x}_0; \theta^*) - \mathbf{f_0^T}(\hat{\theta} - \theta^*) = \varepsilon_0 - \mathbf{f_0^T}(\hat{\theta} - \theta^*) \tag{2.7}$$

Levando-se em conta que $\hat{\theta}$ e ε_0 são estatisticamente independentes, o valor esperado do resíduo ($E[y_0 - \hat{y}_0]$) para uma nova observação y_0 bem como a variância deste valor esperado ($\mathrm{var}[y_0 - \hat{y}_0]$) podem ser representados pelas equações abaixo:

$$E[y_0 - \hat{y}_0] \approx E[\varepsilon_0] - \mathbf{f_0^T} E[(\hat{\theta} - \theta^*)] \approx 0 \tag{2.8}$$

e

$$\mathrm{var}[y_0 - \hat{y}_0] \approx \mathrm{var}[\varepsilon_0] + \mathrm{var}[\mathbf{f_0^T}(\hat{\theta} - \theta^*)] \approx \sigma^2 + \sigma^2 \mathbf{f_0^T}(\mathbf{F^T F})^{-1} \mathbf{f}_0 \tag{2.9}$$

onde

- $E[z]$ denota a esperança ou valor esperado da variável z;
- $\mathrm{var}[z]$ denota a variância da variável z;
- σ^2 é a variância dos resíduos do modelo e;
- **F** representa a matriz Jacobiana do modelo com dimensões $n \times p$, onde n é o número de exemplos utilizados para estimar os p parâmetros do modelo, e o elemento na posição i, j da matriz é escrito como

$$\frac{\partial f(x_i, \hat{\theta})}{\partial \hat{\theta}_j}$$

É factível dizer então que $y_0 - \hat{y}_0$ possui, assintoticamente, uma distribuição na forma $N(0, \sigma^2[1 + \sigma^2 \mathbf{f_0^T}(\mathbf{F^T F})^{-1}\mathbf{f}_0])$ e que como σ^2 é independente de y_0 e assintoticamente independente de $\hat{\theta}$, ele também é assintoticamente independente de $y_0 - \hat{y}_0$.

A partir destas afirmações, uma distribuição t de Student, assintoticamente válida, pode ser escrita na forma

$$\frac{y_0 - \hat{y}_0}{\sigma\sqrt{1 + \mathbf{f_0^T}(\mathbf{F^T F})^{-1}\mathbf{f}_0}} \sim t_{n-p} \tag{2.10}$$

através da qual é possível afirmar que o real valor da predição de y_0 está, com $100(1-a)\%$ de confiança, localizado no intervalo determinado por:

$$\hat{y}_0 \pm t_{n-p}^{\alpha/2} s[1 + \mathbf{f_0^T}(\mathbf{F^T F})^{-1}\mathbf{f}_0]^{1/2} \tag{2.11}$$

onde s pode ser calculado a partir do estimador não enviesado de σ^2 obtido através da Equação (2.12) e o vetor de parâmetros ideais θ^*, no termo $\mathbf{f}_0$, é substituído pelos parâmetros estimados $\hat{\theta}$.

$$s^2 = \frac{\sum_{i=1}^{n}(y_i - f(x_i; \hat{\theta}))^2}{n - p} \tag{2.12}$$

É importante observar que o IP é construído a partir de dois termos presentes na Equação (2.11): o primeiro é a própria estimativa do modelo de regressão não-linear para uma nova observação x_0; e o segundo é o termo $t_{n-p}^{\alpha/2} s[1 + \mathbf{f_0^T}(\mathbf{F^T F})^{-1}\mathbf{f}_0]^{1/2}$, que define por si só metade da amplitude do IP estimado. O dobro deste termo será referenciado neste trabalho, portanto, como Amplitude do Intervalo de Predição (AIP) e será esta AIP que servirá como indicador do quão precisa é uma resposta do modelo.

Como dito anteriormente, o IP determina uma região na qual se espera, com um determinado nível de confiança, que o valor real da variável de interesse esteja localizado. Uma região grande implica em um maior número de possibilidades para o valor real da variável de interesse e consequentemente em uma menor probabilidade da resposta do modelo estar correta. Do mesmo modo, uma região pequena diminui esse número de possibilidades, aumentando as chances de que os valores de $\hat{y}_0$ e y_0 sejam

equivalentes. É natural portanto aceitar que a AIP sirva como um indicador da precisão das estimativas obtidas a partir de um modelo de regressão.

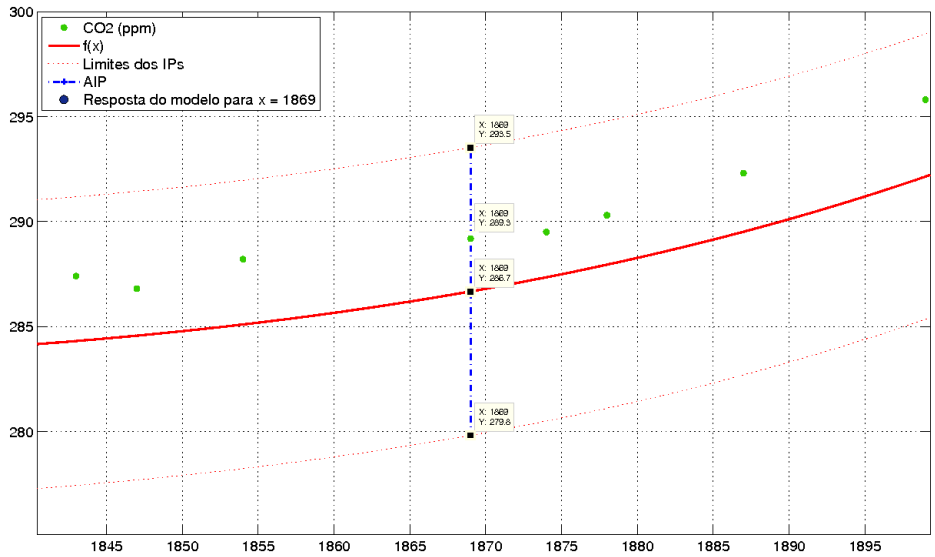

Figura 2.3: Intervalos de predição obtidos a partir do modelo usado no exemplo da Seção 2.1 para um nível de confiança de 95%. A linha em azul destaca a AIP obtida para a estimativa da variável y dada uma observação $x = 1869$.

2.3 Intervalos de predição para RNAs

Uma RNA do tipo MLP (do inglês *Multi Layer Perceptron*) é uma rede neural composta por neurônios agrupados em camadas que se interconectam através de ligações denominadas sinapses. Uma MLP possui necessariamente uma camada de entrada, através da qual dados de entrada são apresentados à rede; uma camada de saída, responsável por fornecer as saídas calculadas; e uma ou mais camadas intermediárias, denominadas camadas escondidas, que são responsáveis por extrair características não observáveis explicitamente dos vetores de entrada (Figura 2.4).

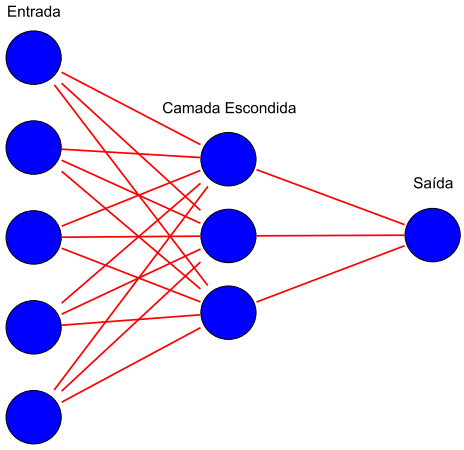

Figura 2.4: Exemplo de uma rede neural *Multi-Layer Perceptron*. Os pesos sinápticos que conectam os neurônios estão em vermelho.

Particularmente, as redes MLP do tipo *Feed Forward* abordadas neste trabalho apresentam um fluxo de informação que se inicia nos neurônios da camada de entrada, passando pelos neurônios da(s) camada(s) escondida(s) (também chamados de unidades escondidas) até finalmente chegar na camada de saída, sem que haja neste processo qualquer tipo de comunicação entre neurônios pertencentes a uma mesma camada da rede.

Segundo [Haykin 1998] em uma rede MLP cada neurônio implementa uma função de ativação não-linear cujo parâmetro de entrada é o somatório ponderado de todas as sinapses conectadas àquele neurônio, formando o chamado campo local induzido. Para um neurônio j qualquer da rede este parâmetro pode ser dividido em dois conjuntos de parâmetros: as respostas dos k neurônios pertencentes à camada anterior e que estão conectados a j, $\mathbf{x}_j = (x_1, x_2, ..., x_k)$, e os pesos das sinapses que conectam tais neurônios $\mathbf{w}_j = w_1, w_2, ...w_k$. Desta forma pode-se escrever a função de ativação do neurônio como:

$$o_j = f(\mathbf{x}_j; \mathbf{w}_j) \tag{2.13}$$

Considerando-se, por exemplo, uma rede MLP simples com uma única camada escondida (além das de entrada e saída) e um neurônio em cada uma das camadas e assumindo ainda que os neurônios da camada escondida e da camada de saída implementem uma função logística, comumente usada nestes tipos de rede, e o neurônio da

camada de entrada sirva somente como unidade de alimentação da rede, as saídas de tais neurônios podem ser escritas da seguinte forma:

$$o_1 = \mathbf{x}_1$$
$$o_2 = \frac{1}{1 + e^{-y_1 \mathbf{w}_2}}$$
$$o_3 = \frac{1}{1 + e^{-y_2 \mathbf{w}_3}}$$

onde o_1, o_2 e o_3 correspondem às funções de saída do neurônio da camada de entrada, escondida e de saída respectivamente. Fica evidente então que a saída da rede, equivalente à saída o_3, pode ser descrita como uma função $\hat{y} = f(\mathbf{x}; \mathbf{w})$ cujos parâmetros $\mathbf{x}$ e $\mathbf{w}$ representam o vetor de entradas apresentado à rede e o vetor de pesos sinápticos da RNA respectivamente.

Fazendo uma extrapolação deste exemplo simples para redes MLP com um número qualquer de neurônios na camada escondida, uma analogia entre modelos de regressão não-linear e redes neurais pode ser estabelecida. Assim como um modelo de regressão, o objetivo de uma RNA é estabelecer uma relação entre variáveis aleatórias que melhor retrate o comportamento da variável de interesse em função das variáveis independentes. A resposta da rede, ou *saída* como é mais comumente chamada, é produto da transformação não-linear realizada pela rede, que tem como parâmetros um vetor $\mathbf{x}$, também chamado de *entrada*, e o vetor de pesos sinápticos $\mathbf{w}$. Estes pesos são combinados de maneira não-linear, através das funções de ativação dos neurônios, gerando como resultado final a saída da RNA.

Inicialmente os pesos de uma rede neural possuem valores aleatórios, que vão sendo gradativamente modificados a fim de que a relação entre as entradas e a variável de interesse possa ser representada de forma aceitável. A esta etapa de ajuste dos pesos sinápticos dá-se o nome de treinamento e durante esta fase um conjunto de treinamento, formado por vetores de entrada e os respectivos valores da variável de interesse (denominados *alvos*), é utilizado da seguinte maneira: uma entrada $\mathbf{x}_i$ é apresentada à rede e a saída da mesma $\hat{y}_i$ é comparada com o valor correspondente y_i da variável de interesse; a diferença entre a saída da rede e o valor da variável é então utilizada para

ajustar os pesos da rede neural. Isto é feito até que todas as entradas do conjunto de treinamento tenham sido apresentadas à rede e as respectivas diferenças tenham sido calculadas, quando isto ocorre diz-se que uma época terminou. O processo se repete com os últimos valores de $\mathbf{w}$ obtidos até que se verifique que a rede atingiu um estágio bom de aprendizado ou até que um número máximo de épocas tenha sido atingido. Esta verificação normalmente é feita utilizando-se um indicador global de desempenho da rede como o *Erro Médio Quadrático* (EMQ).

Estes passos descrevem de forma geral o algoritmo de *backpropagation* ou *retropropagação de erro*, cuja origem do nome está ligada ao fato dos erros (as diferenças entre os valores de y e $\hat{y}$) serem propagados, desde a camada de saída até a camada de entrada, para todos os pesos sinápticos da rede neural a fim de que estes possam ser modificados. Como visto em [Haykin 1998], assim como ocorre na estimação dos parâmetros de um modelo de regressão não-linear, a estimação dos pesos sinápticos de uma rede neural pode ser tratada como um problema de otimização numérica, o que permite a aplicação, durante a fase de treinamento, de algoritmos de otimização de parâmetros como os apresentados no Apêndice A.

Pode-se considerar, portanto, que uma rede MLP seja um tipo de modelo de regressão não-linear onde os pesos sinápticos $\mathbf{w}$ desempenham o papel dos parâmetros θ e os resíduos ε são representados pelas diferenças entre os alvos y e as saídas da rede $\hat{y}$. Por conseguinte, a teoria para estimação de intervalos de predição, apresentada na Seção 2.2, pode ser aplicada diretamente nesses tipos de rede [Dybowski and Roberts 2001].

Como o vetor de parâmetros, neste caso, equivale aos pesos sinápticos $\mathbf{w}$, tem-se que o termo $\mathbf{f_0^T}$ é representado agora por

$$\mathbf{f_0^T} = \left(\frac{\partial f(\mathbf{x}_0; \mathbf{w}^*)}{\partial \mathbf{w}_1^*}, \frac{\partial f(\mathbf{x}_0; \mathbf{w}^*)}{\partial \mathbf{w}_2^*}, \cdots, \frac{\partial f(\mathbf{x}_0; \mathbf{w}^*)}{\partial \mathbf{w}_p^*} \right)$$

onde $\mathbf{x}_0$ é uma nova entrada, não utilizada durante o treinamento, fornecida à rede; e $\mathbf{w}^*$ denota os pesos sinápticos ideais que representam o sistema o qual se deseja modelar. Pelo mesmo motivo a matriz Jacobiana tem cada um dos seus elementos i, j escritos como

$$\frac{\partial f(x_i, \hat{\mathbf{w}})}{\partial \hat{\mathbf{w}}_j}$$

e o estimador não enviesado da variância dos resíduos é dado por:

$$s^2 = \frac{\sum_{i=1}^{n}(y_i - f(\mathbf{x}_i; \hat{\mathbf{w}}))^2}{n - p} \tag{2.14}$$

onde $f(\mathbf{x}_i; \hat{\mathbf{w}})$ representa o modelo obtido com a rede neural e $\hat{\mathbf{w}}$ remete ao vetor de pesos sinápticos estimados no processo de treinamento.

Em [Chryssolouris et al. 1996] tal abordagem foi adotada com sucesso quando aplicada na indústria automotiva, demonstrando a viabilidade e praticidade deste método. A seguir uma revisão bibliográfica dos trabalhos já realizados na área é feita com o intuito de se justificar a escolha deste método como foco no presente trabalho.

2.4 Revisão bibliográfica

Na literatura relacionada ao tema deste trabalho, encontram-se diferentes técnicas que se propõem a disponibilizar uma maneira de obter intervalos de predição quando no uso redes neurais. Como dito na Seção 1.2, tais técnicas podem ser divididas em dois grupos: técnicas de caráter global, as quais fazem uso de informações mais generalizadas no que diz respeito aos dados de treinamento; e técnicas de caráter local que tentam extrair uma ou mais dessas informações de forma mais individualizada.

Em [Wong et al. 2002], tem-se um exemplo de técnica pertencente ao primeiro grupo. A proposta apresentada (aplicada na caracterização de reservatórios) consiste em mapear os valores dos alvos relacionados às saídas da rede neural em classes (ou variáveis linguísticas) previamente definidas a partir de histogramas ou do conhecimento extraído de especialistas da área para a qual a rede neural está sendo modelada, transformando a tarefa de predição da RNA em uma trabalho de classificação. Esta divisão em grupos pode ser tanto de natureza booleana, com um determinado valor podendo pertencer somente a uma classe, como de natureza *fuzzy* a qual permite a pertinência dos dados a várias classes simultaneamente. Neste trabalho o autor opta por utilizar uma abordagem *fuzzy* com as saídas da rede neural passando a representar agora os graus de pertinência dos vetores de entrada em cada classe ao invés do valor da variável de interesse.

Para determinar a confiança da classificação realizada pela rede neural para um determinado vetor de entrada, a entropia dos valores de pertinência obtidos a partir deste vetor é calculada. Um valor de entropia alto significa que a rede classificou um determinado vetor de entrada de forma aproximadamente semelhante para todas as classes, o que consequentemente indica um nível de confiança baixo para o resultado. Já um valor de entropia baixo indica que foi possível determinar mais seguramente que tal vetor pertence a uma determinada classe linguística, i.e., o resultado obtido é mais confiável. Além deste indicador, uma transformação reversa dos graus de pertinência obtidos também é proposta. Através desta transformação os valores máximo e mínimo que uma determinada observação da variável de interesse pode assumir e o valor que teria sido estimado caso a rede estivesse sendo aplicada em um trabalho de predição são obtidos, definindo então um intervalo de predição para o valor real desejado.

Apesar da facilidade de implementação do método, o qual não requer alterações nos tipos de estrutura ou nos algoritmos de treinamento normalmente utilizados com redes neurais, não fica estabelecido um nível de confiança para os intervalos de predição obtidos. Além disso, tal metodologia não leva em consideração questões como a adequação dos pesos sinápticos da rede neural aos dados de treinamento, ruído e densidade de dados disponíveis. Por fim, a construção das classes linguísticas é realizada de maneira subjetiva tornando os resultados obtidos dependentes das interpretações feitas pelo responsável por criar as mesmas.

Outros trabalhos de caráter global optaram por utilizar métodos com um melhor embasamento na Estatística para calcular os intervalos de predição. No trabalho apresentado em [Veaux et al. 1998] aborda-se a situação em que há uma pequena quantidade de dados disponíveis para o treinamento da rede neural. Nestas circunstâncias, executar o algoritmo de treinamento das redes até a sua convergência irá fazer com que ocorra um *overfitting* da RNA e consequentemente a variância dos resíduos, um dos fatores que compõem o cálculo dos IPs, acaba por ser subestimada.

A fim de que isto seja evitado, técnicas como regularização dos pesos sinápticos e *early stopping* normalmente são empregadas, reduzindo o número de parâmetros que agem de forma significativa na RNA e tornando a resposta da mesma mais suave.

Com um número de parâmetros efetivos menor do que a quantidade de pesos sinápticos, o cálculo de IPs da forma como é realizado em [Chryssolouris et al. 1996] superestima a variância dos resíduos, resultando em intervalos muito conservadores e que portanto não retratam a real confiabilidade das respostas da RNA.

Uma modificação nas fórmulas utilizadas em [Chryssolouris et al. 1996] para o cálculo dos IPs é, então, apresentada juntamente com uma maneira de se calcular o número de parâmetros efetivos da rede, para que desta forma os intervalos retratem com fidelidade o nível de confiança desejado das respostas de uma rede neural treinada com o algoritmo *Weight-decay*.

Apesar dos bons resultados apresentados em [Chryssolouris et al. 1996] e [Veaux et al. 1998], a teoria empregada por eles requer que certas restrições, raramente satisfeitas em aplicações reais, sejam atendidas. A rede neural tem de ser treinada até a convergência do algoritmo de estimação dos parâmetros (excetua-se neste caso o método apresentado em [Veaux et al. 1998]); não deve haver erros de medição das variáveis de entrada; os resíduos devem ser identicamente e independentemente distribuídos (i.i.d.); e, por fim, tais resíduos devem estar distribuídos normalmente com média zero e variância constante. Além disso, em nenhum dos dois trabalhos citados, a distribuição dos dados de entrada é levada em consideração. A não-conformidade do modelo obtido com tais suposições pode ocasionar a estimação de IPs que não reflitam corretamente o nível de confiabilidade que deve ser atribuído às respostas da RNA.

A proposta feita em [Shao et al. 1997], de caráter local e voltada para redes *Multi-Layer Perceptron* (MLP), tem como meta a solução do problema decorrente de uma distribuição não-uniforme dos dados de entrada do conjunto de treinamento, isto é, densidades de dados maiores ou menores em determinadas regiões do espaço de entrada. Tais disparidades na densidade do domínio da rede neural deveriam ser retratadas por intervalos de predição maiores em regiões com baixa densidade de dados, e intervalos menores em zonas em que a quantidade de observações é maior.

Para alcançar este objetivo, uma segunda rede neural, além da utilizada na modelagem do sistema de interesse, é proposta a fim de que um estimador local da densidade dos dados de entrada possa ser obtido. Tal rede é chamada de *wave-net* pelo fato

das funções de ativação dos neurônios serem *wavelets*, uma família de funções largamente utilizada para expressar e aproximar outras funções [Shao et al. 1997]. A densidade estimada em função de um determinado dado de entrada é então incorporada ao cálculo do IP através de um coeficiente inversamente proporcional à saída da *wave-net*.

Em [Leonard et al. 1992] uma nova arquitetura de rede capaz de calcular IPs para suas respostas é apresentada. Esta arquitetura, chamada de *VI-net (validity index network)*, é uma modificação da arquitetura usual de redes de função de base radial (RFBR), onde além das unidades escondidas habituais, novas unidades são adicionadas para realizar o cálculo dos intervalos de predição. Graças à capacidade inerente das RFBRs de particionar o espaço de entrada, os IPs obtidos a partir da VI-net são capazes de refletir diferentes taxas de ruído nas variáveis de saída. Ainda aproveitando-se desta peculiaridade das RFBRs, dois indicadores são também calculados: o primeiro é responsável por informar quando a rede está realizando uma extrapolação; e o segundo fornece uma estimativa da densidade de dados da região em que se localiza o vetor de entrada apresentado à rede, utilizando para isto o método das janelas de Parzen.

A despeito de se mostrar eficaz na detecção de ruídos na variável de saída, o modo como os intervalos de predição são calculados em [Leonard et al. 1992] restringe-se às RFBRs, não sendo possível aplicá-lo nas redes MLP, para as quais a relação entre neurônios da camada escondida e regiões no espaço de entrada não existe.

Buscando obter um método com este tipo de capacidade para redes MLP, *Chinman & Ding* propõem em [Chinman and Ding 1998] uma abordagem semelhante à apresentada em [Leonard et al. 1992]. Para tal, um *Mapa Auto-Organizável de Kohonen* é aplicado ao conjunto de treinamento da rede MLP, dando origem a regiões mutuamente exclusivas no espaço de entrada, dentro das quais a variância dos resíduos é constante, mas variável de região para região. Assim, considerando que os neurônios da camada de saída do Mapa de Kohonen exercem o mesmo papel que as unidades escondidas das RFBRs, os IPs são calculados de forma análoga ao que é feito em [Leonard et al. 1992].

Deve-se notar, todavia, que da mesma forma que em [Leonard et al. 1992], a proposta em [Chinman and Ding 1998] utiliza equações de cálculo de IPs para

modelos de regressão linear. O uso destas equações se justifica para as RFBRs, pois todo o cálculo envolvido na estimação dos intervalos tem como foco os pesos sinápticos que ligam as unidades escondidas da rede à camada de saída, os quais podem ser obtidos através de uma simples regressão linear. O mesmo não ocorre no treinamento de redes MLP, as quais são um tipo de modelo de regressão não-linear, o que torna mais sensato o uso de métodos voltados para este tipo de modelo como o apresentado em [Chryssolouris et al. 1996].

Outra proposta pertencente ao grupo de técnicas de caráter local é apresentada em [Nix and Weigend 1995]. A abordagem desse trabalho consiste na adição de um novo neurônio de saída ligado a uma camada oculta adicional, composta por neurônios ligados tanto aos neurônios da camada de entrada como às unidades escondidas tradicionais de uma rede MLP. A estrutura adicional na arquitetura da rede age como um estimador da variância do resíduo em função de um vetor de entrada específico, possibilitando que os IPs estimados para as respostas da rede neural possam refletir a taxa de ruído nas variáveis de saída. Além disso, mostra-se que graças a esta modificação a generalização da rede neural é melhorada de forma significativa.

Embora o método proposto em [Nix and Weigend 1995] obtenha bons resultados e melhore a performance da RNA, a modificação na arquitetura da rede aumenta a complexidade do aprendizado, pois requer que um conjunto adicional de parâmetros seja estimado. A rede neural desejada somente é obtida após um processo que consiste de 3 fases, as quais são necessárias para evitar que o efeito regularizador introduzido pelos novos parâmetros acabe por prejudicar o aprendizado da mesma.

Abordagens globais, calcadas em técnicas de *bootstrapping* apresentadas em [Heskes 1997] e [Lebaron and Weigend 1994], são atestadas como tendo ótimas performances, porém seus custos computacionais são muito elevados, principalmente quando se lida com conjuntos de dados e redes neurais muito grandes [Dybowski and Roberts 2001], [Heskes 1997], [Papadopoulos et al. 2001].

Por fim, redes Bayesianas também têm sido empregadas com sucesso no cálculo de intervalos de predição. Tais redes podem ser incluídas tanto no grupo de técnicas de caráter global como as de caráter local, uma vez que as mesmas podem ser

modificadas para calcularem a variância dos resíduos em função das entradas apresentadas [Papadopoulos et al. 2001]. A implementação destas abordagens, contudo, é mais trabalhosa do que quando se usam redes treinadas através de métodos de *Estimação de Máxima Verossimilhança* (EMV) como feito em [Chryssolouris et al. 1996]. A estimação da incerteza dos parâmetros obtidos em redes Bayesianas somente são válidas quando tais cálculos são feitos com uma exatidão razoável, sem o uso de aproximações. No caso da EMV, tanto a estabilidade como a velocidade de convergência do algoritmo permitem uma fácil implementação do método [Dybowski and Roberts 2001].

Buscando-se então melhorar o método proposto em [Chryssolouris et al. 1996] de forma a não aumentar demasiadamente o seu custo computacional, uma extensão para esta abordagem será apresentada no capítulo seguinte.

Capítulo 3

Proposta para Cálculo do Intervalo de Predição

Este capítulo se inicia com uma situação hipotética, através da qual se deseja expor as consequências da aplicação do método delta quando na presença de resíduos com variância não-constante. O capítulo continua com uma explicação dos aspectos pertencentes à proposta feita em [Leonard et al. 1992] que serviram como inspiração para a abordagem deste trabalho, e então a extensão ao método proposto em [Chryssolouris et al. 1996] é apresentada.

3.1 Método delta aplicado a dados com ruído não-uniforme

No capítulo anterior, o método delta, usado no cálculo de intervalos de predição para estimações de modelos de regressão não-linear, foi apresentado na Seção 2.2 como uma solução viável para se obter um indicador de confiabilidade das saídas fornecidas por redes *Multi-Layer Perceptron*.

Apesar dos bons resultados obtidos em [Chryssolouris et al. 1996], tal método exige que as suposições citadas na Seção 1.2 sejam feitas para que os IPs estimados reflitam de forma coerente o nível de confiança para o qual eles foram calculados.

A suposição de que a variância dos resíduos seja constante é particu-

larmente forte [Chinman and Ding 1998] e para efeitos da proposta apresentada neste trabalho, será considerado que tal suposição não pode ser garantida na coleta de dados em aplicações reais.

Os resíduos nada mais são que os erros da rede neural referentes aos dados de treinamento, e o comportamento dos mesmos é determinado, dentre outros fatores por erros de medida da variável de interesse e pelo erro associado ao modelo [Papadopoulos et al. 2001]. Estes dois tipos de erro serão denominados daqui em diante como ruído, uma vez que ambos são responsáveis por fazer com que a verdadeira natureza do sistema modelado não seja captada pela RNA, gerando então os resíduos já mencionados.

Considera-se nesta dissertação que as demais fontes de incertezas (erros grosseiros de modelagem da rede neural e distribuição não-uniforme dos dados de entrada) não estão presentes nos sistemas aqui tratados, tornando o ruído a principal fonte causadora de resíduos.

Ao supor que a variância dos resíduos σ^2 é constante, os IPs estimados para as respostas da RNA são afetados de maneira que os mesmos refletem o ruído existente apenas de forma geral, não distinguindo explicitamente regiões com baixa taxa de ruído de regiões muito ruidosas.

Em [Leonard et al. 1992] um exemplo hipotético, onde a variância dos resíduos não é constante, é usado como parte dos experimentos executados. Neste exemplo um ruído artificial, calculado em função do vetor de entrada $\mathbf{x}$, é adicionado ao sistema de interesse com o intuito de simular resíduos desta natureza. O mesmo sistema será utilizado, a seguir, com a finalidade de melhor demonstrar os efeitos causados pela suposição de que a variância residual é constante quando este fato não é verdade.

No referido exemplo o sistema de interesse, para o qual se deseja obter uma rede neural que retrate a relação entre duas variáveis x e y, é definido pela equação a seguir:

$$y(x) = 0.5sen(1.5\pi x + \pi/2) + 2.0 + \nu \tag{3.1}$$

onde

- x representa a variável independente;
- y representa a variável de interesse;
- e ν é um ruído Gaussiano com desvio-padrão denotado por:

$$\sigma_\nu = 0.045 + 0.04x \tag{3.2}$$

Para tal sistema, 200 pares de observações (x, y) foram amostrados, com os valores de x variando uniformemente dentro do intervalo $[-1, +1]$. A Figura 3.1 mostra a curva resultante de quando y é plotado em função de x. É interessante notar que o ruído presente na variável y aumenta proporcionalmente ao valor da variável independente dando origem a uma região pouco ruidosa para valores de x próximos a -1 e a uma zona com alta taxa de ruído para valores de x próximos a $+1$. Tal comportamento pode ser visto mais explicitamente na Figura 3.2.

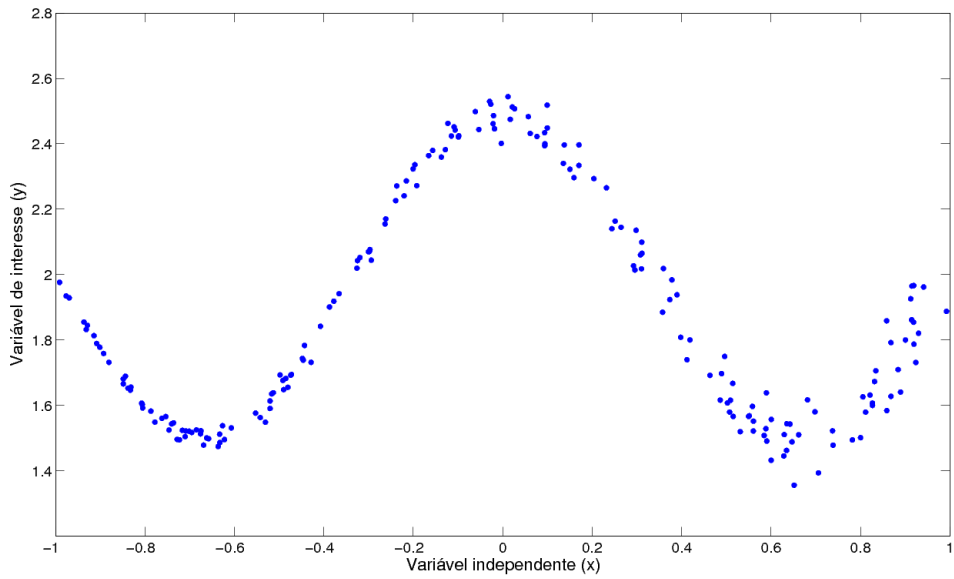

Figura 3.1: Gráfico mostrando o comportamento da Eq. (3.1). É possível notar que o ruído nas observações de y aumenta à medida que o valor de x cresce.

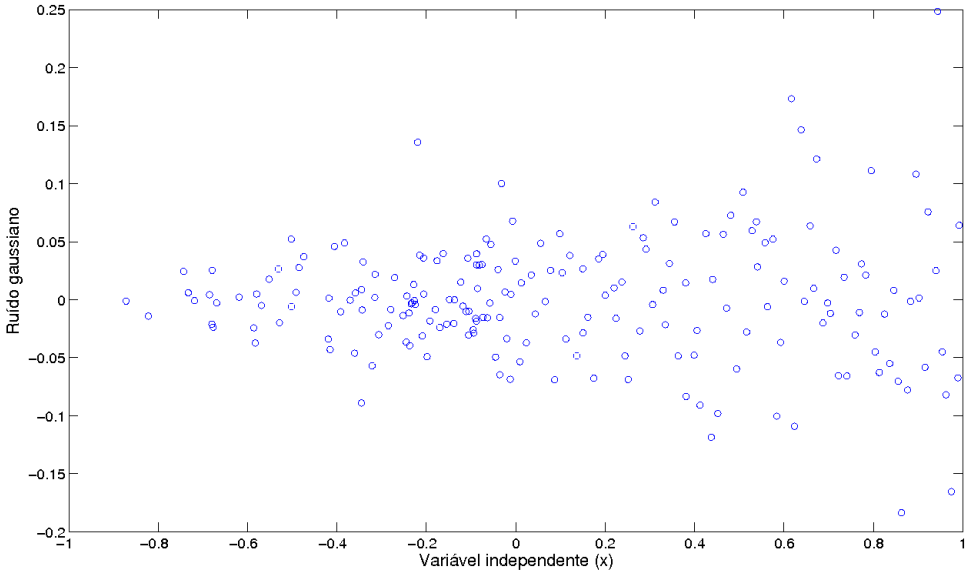

Figura 3.2: Ruído gaussiano em função do valor da variável independente.

Uma RNA com 5 neurônios na camada escondida, 1 neurônio na camada de entrada e 1 neurônio na camada de saída (Figura 3.3) foi treinada utilizando este conjunto de dados até que um modelo que representasse satisfatoriamente o sistema fosse obtido.

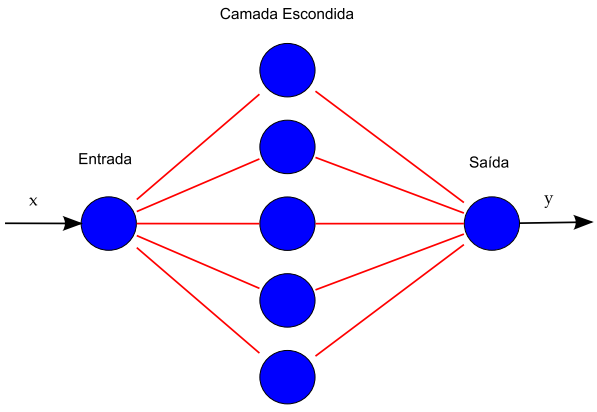

Figura 3.3: Arquitetura da RNA utilizada com o método delta.

Após esta etapa, as 200 observações da variável x foram apresentadas como entradas para a rede neural a fim de que a mesma fornecesse estimativas para o vetor de observações $(y_1, y_2, \cdots, y_{200})$ juntamente com os IPs, para um grau de confiança de 95%, onde se acredita que estas observações estejam localizadas.

Como pode ser percebido através das Figuras 3.4 e 3.5, apesar do ruído na variável y crescer cada vez mais à medida que o valor de x aumenta, os IPs obtidos pelo método delta não refletem de forma significativa este comportamento.

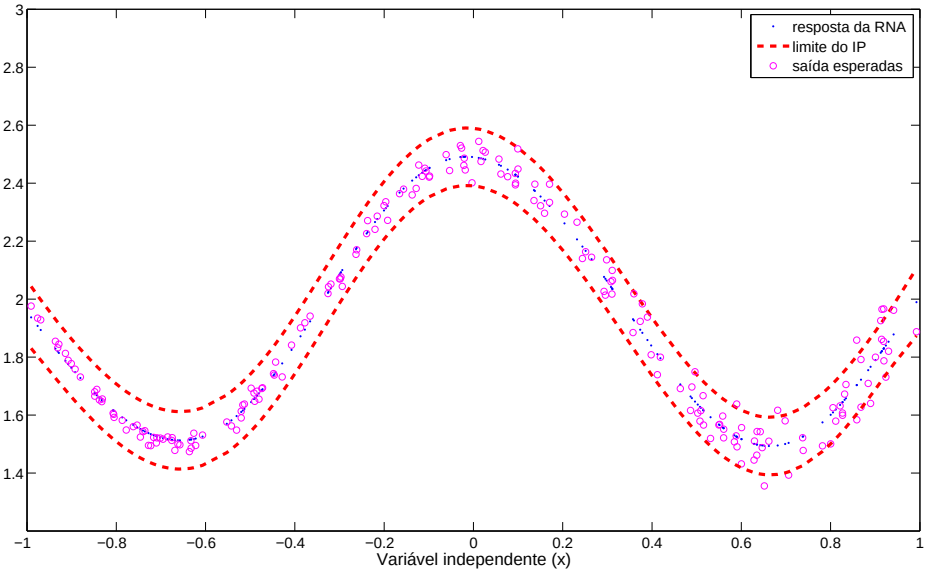

Figura 3.4: Estimações da RNA , IPs e observações da variável y em função da variável independente x.

Tomando, por exemplo, o valor $x_0 = -0.8057$, localizado em uma região com baixa taxa de ruído, obtém-se uma AIP de valor 0.1986. Tomando-se outra observação, desta vez em uma região com mais ruído, de valor 0.8054, tem-se uma AIP estimada de 0.2003. A diferença de 0.0017 entre as duas AIPs é (para este sistema em particular) insignificante, dando a falsa ilusão de que as estimativas da RNA para ambas observações são igualmente precisas. Porém, há uma probabilidade maior da rede neural fornecer estimativas y_0 mais precisas para dados de entrada localizados em regiões com baixa taxa de ruído, onde o aprendizado da mesma se deu com mais eficiência, do que

para valores da variável regressora que estão em zonas mais ruidosas. Este fato deveria ser retratado por IPs mais estreitos do que aqueles obtidos para valores da variável independente que estão em regiões com alta taxa de ruído.

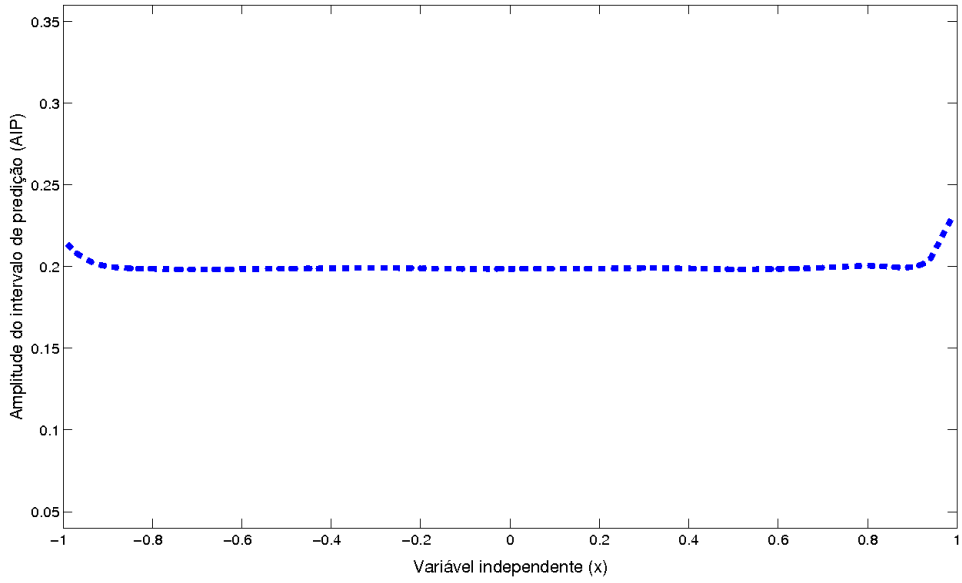

Figura 3.5: Valores da amplitude dos IPs em função da variável regressora x.

Em outras palavras, não é possível inferir se uma determinada resposta fornecida pela RNA está em uma região cuja taxa de ruído é alta ou baixa, o que acaba por comprometer as suposições feitas a respeito da precisão das respostas da rede neural.

3.2 Método Delta Estendido

Com o intuito de contornar a limitação do método delta, apresentada na Seção 3.1, uma extensão para o mesmo é proposta. Tal extensão será denominada daqui em diante como Método Delta Estendido (MDE) e o método delta será referenciado como Método Delta Clássico (MDC).

Para que IPs sejam obtidos a partir das respostas de uma rede neural de forma mais precisa é interessante que a variância dos resíduos possa ser calculada como uma função dos dados de entrada da RNA [Papadopoulos et al. 2001].

A VI-net (*Validity Index network*), apresentada em [Leonard et al. 1992], é capaz de estimar a variância dos resíduos desta forma graças à uma peculiaridade das redes de função de base radial (RFBR), arquitetura na qual a VI-net se baseia.

Uma RFBR é uma arquitetura de rede neural que consiste basicamente de 3 camadas de neurônios arranjadas da mesma maneira que uma rede MLP, mas diferentemente das redes MLP as sinapses que partem da primeira camada para a segunda não possuem pesos. Consequentemente os neurônios dessa camada acabam apenas por repassar os dados de entrada apresentados à rede para a camada escondida, sem aplicar nenhum tipo de transformação sobre os mesmos. A principal diferença existente entre essas duas arquiteturas, contudo, se deve ao tipo de função de ativação que os neurônios da camada escondida de cada uma implementa. Enquanto que nas MLPs as funções implementadas são na sua maioria funções logísticas, sigmoidais ou lineares, nas RFBRs os neurônios da camada escondida (ou unidades FBR) possuem funções similares à função gaussiana multivariada de densidade (Figura 3.6), que é descrita pela equação:

$$a_h(\mathbf{x}) = e^{-\|\mathbf{x}-\mathbf{x}_h\|^2/\sigma_h^2} \tag{3.3}$$

onde:

- a_h representa a saída do neurônio h da camada escondida dado um vetor de entradas $\mathbf{x}$;
- $\mathbf{x}_h$ é a posição do centro da gaussiana no espaço de entrada e;
- σ_h é chamada de abertura da gaussiana, a qual determina até que distância, no espaço de entrada, a unidade h terá um nível de influência significativo.

Como resultado, cada função implementada pelas unidades FBR da rede neural possui influência apenas em uma região específica do espaço de entrada, a qual é definida pelos parâmetros $\mathbf{x}_h$ e σ_h. As unidades FBR podem ser vistas, portanto, como elementos delimitadores de regiões, ou vizinhanças, no domínio utilizado como espaço de entrada da rede.

Devido a este conceito de vizinhanças no espaço de entrada inerente às RFBRs, é possível tratar a variância dos resíduos como sendo constante apenas dentro de uma vizinhança delimitada por uma unidade FBR específica, e como variável de vizinhança para vizinhança [Leonard et al. 1992]. Considerando que o ruído existente seja o principal causador dos resíduos do modelo, é possível para a VI-net, portanto, calcular IPs sensíveis à distribuição deste ruído.

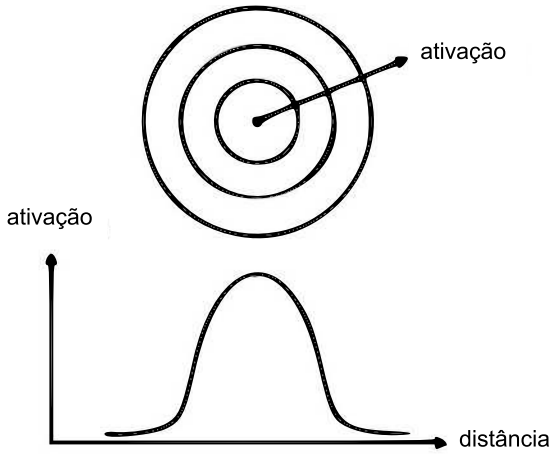

Figura 3.6: Função de ativação de uma unidade escondida de uma RFBR.

Este conceito, contudo, não pode ser aplicado para as redes *Multi-Layer Perceptron*. As ativações dos neurônios da camada escondida das RFBR são feitas baseadas em uma métrica de distância entre o vetor de entrada e os centros destes neurônios. Essas ativações acabam por criar contornos constantes no formato de hiperesferas de tamanho finito, dando origem às vizinhanças já citadas anteriormente. As unidades escondidas de uma rede MLP, por outro lado, definem hiperplanos infinitos e que por sua vez dão origem a regiões semi-infinitas de ativações significativas [Leonard et al. 1992].

Neste trabalho o conceito de vizinhanças das RFBR foi aplicado nas redes MLP com o auxílio de um artifício. Tal artifício consiste, basicamente, em se criar as vizinhanças no espaço de entrada utilizando uma estrutura à parte da arquitetura da rede neural.

Após o treinamento da rede MLP, processo em que os pesos sinápticos são estimados, um algoritmo de agrupamento (como k-médias) é utilizado para classificar os n dados de entrada utilizados no treinamento da rede neural em um número pré-definido de *clusters*, denotado aqui por C. Estes *clusters* irão delimitar vizinhanças no espaço de entrada com diferentes variâncias residuais.

A partir disto, o grau de pertinência de um vetor de entrada $\mathbf{x}$ para cada uma das vizinhanças pode ser obtido utilizando-se o nível de ativação deste vetor em cada *cluster*. Assumindo que gaussianas esféricas tenham sido usadas no processo de agrupamento, o nível de ativação de um vetor $\mathbf{x}$ para um *cluster* c pode ser descrito matematicamente por:

$$a_c(\mathbf{x}) = e^{-\frac{\|\mathbf{x} - \mathbf{x}_c\|^2}{\sigma_c^2}} \tag{3.4}$$

onde:

- $\mathbf{x}_c$ representa o centro do *cluster* c e;
- σ_c é a abertura da gaussiana correspondente ao *cluster* c;

Desta forma, quanto mais próximo $\mathbf{x}$ estiver do centro de um *cluster*, maior será o grau de pertinência deste vetor à vizinhança determinada por este *cluster* (Figura 3.7).

Após o processo de agrupamento dos dados de entrada, a variância dos resíduos de cada *cluster* pode ser calculada através da Equação (3.5).

$$s_c^2 = \frac{\sum_{i=1}^{n} a_c(\mathbf{x}_i) * \hat{\varepsilon}_i^2}{n_c - p} \tag{3.5}$$

onde:

- s_c^2 é a variância dos resíduos estimada para o *cluster* c;
- $a_c(\mathbf{x}_i)$ é dada conforme a Equação (3.4);
- $\hat{\varepsilon}_i = (y_i - f(\mathbf{x}_i, \hat{\mathbf{w}}))$ e;

- n_c é o número total de dados de entrada pertencentes ao *cluster* c, o qual é representado pela equação

$$n_c = \sum_{i=1}^{n} a_c(\mathbf{x}_i)$$

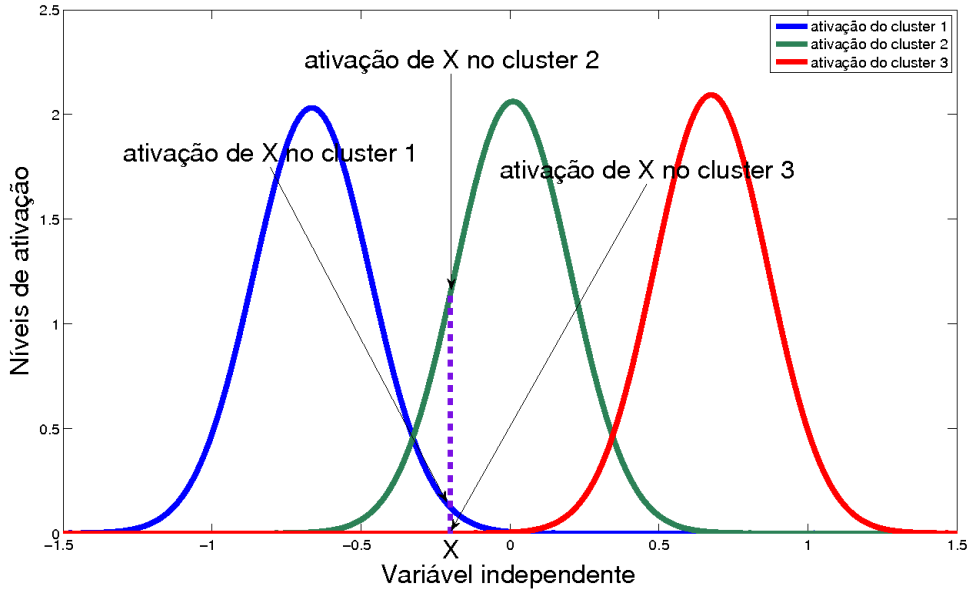

Figura 3.7: Ativações de uma entrada $\mathbf{x}$ para 3 gaussianas hipotéticas. Nesta situação, $\mathbf{x}$ possui um grau de pertinência maior para a vizinhança delimitada pelo *cluster* 2.

Com os graus de pertinência às vizinhanças e as variâncias dos resíduos de cada *cluster* em mãos, uma nova variância residual pode ser obtida através do cálculo da média de todas as s_c^2 ponderadas pelos respectivos níveis de ativação.

$$s_a^2(\mathbf{x}) = \frac{\sum_{c=1}^{C} a_c(\mathbf{x}) s_c^2}{\sum_{c=1}^{C} a_c(\mathbf{x})} \tag{3.6}$$

O termo $s_a^2(\mathbf{x})$, que representa a nova variância residual, será chamado daqui em diante de *variância dos resíduos ajustada* e seu valor é calculado em função do vetor de observações $\mathbf{x}$ apresentado como entrada à rede MLP. Quanto mais próximo do centro de um *cluster* c o vetor de entrada $\mathbf{x}$ estiver, maior será a contribuição da variância residual deste *cluster* no cálculo de $s_a^2(\mathbf{x})$. Ao mesmo tempo, as variâncias residuais dos demais *clusters* agem como fatores de suavização, impedindo que as variâncias dos

resíduos ajustadas referentes a vetores de entrada localizados nos limites de vizinhanças distintas sejam demasiadamente díspares.

Finalmente, tendo-se calculado a variância dos resíduos ajustada, $s_a^2(\mathbf{x}_0)$, para uma nova observação $\mathbf{x}_0$, o cômputo do IP, onde se espera que a observação y_0 esteja contida com $100(1-a)\%$ de confiança, é feito por:

$$\hat{y}_0 \pm t_{n-p}^{\alpha/2} s_a(\mathbf{x}_0)(1 + f_0^T (F^T F)^{-1} f_0)^{\frac{1}{2}} \tag{3.7}$$

Desta forma, a divisão do espaço de entrada em vizinhanças com variâncias residuais próprias provê uma resolução maior sobre as propriedades dos dados utilizados, permitindo que uma estimativa da variância dos resíduos com um caráter mais localizado seja obtida, diferentemente do que acontece com o MDC, onde a variância residual é estimada para todo o domínio utilizado pela rede neural, e por isso retrata a dispersão dos resíduos de uma forma generalizada apenas.

No próximo capítulo os experimentos realizados utilizando o MDE e o MDC serão apresentados juntamente com a análise dos seus resultados.

Capítulo 4

Experimentos e Análise dos Resultados

Com o intuito de comparar as performances do MDE e do MDC alguns experimentos foram realizados. Os experimentos contidos neste capítulo estão divididos em duas categorias: hipotéticos, onde a função que representa o sistema a ser modelado pela rede neural e o comportamento do ruído são conhecidos; e sísmicos, onde a função que representa o sistema bem como o ruído existente não são conhecidos.

Os objetivos específicos destes experimentos são:

1. Mostrar que o método proposto neste trabalho é capaz de retratar diferentes taxas de ruído através das AIPs;

2. Demonstrar que através do MDE é possível obter IPs que retratem o nível de confiança para o qual foram estimados.

Na primeira seção deste capítulo a metodologia usada na execução e coleta de resultados dos experimentos será apresentada, seguida então pela descrição dos mesmos. Por fim os resultados de tais experimentos serão expostos e discutidos.

4.1 Metodologia

Para cada experimento será utilizada mais de uma rede MLP, cada uma treinada a partir de um conjunto de dados de treinamento distinto, com os valores das

variáveis independentes sendo escolhidos de forma aleatória. Como o objetivo deste trabalho não envolve a obtenção de uma rede neural que melhor retrate o sistema em questão, mais de uma rede neural será utilizada a fim de que possíveis variações no conjunto de pesos sinápticos estimados, não influenciem de forma significativa a análise dos métodos de estimação de intervalos de predição.

Após o treinamento das redes, a análise do desempenho dos métodos será feita através de IPs estimados para um nível de confiança de 95%, a partir de conjuntos de dados diferentes dos usados para se estimar os pesos sinápticos das RNAs. Tais conjuntos serão denominados de conjuntos de teste, e se fazem necessários para que o MDC e o MDE sejam analisados quando dados não vistos durante o treinamento das redes neurais sejam apresentados às mesmas.

Em vistas de se averiguar a segunda meta dos experimentos, um indicador quantitativo, normalmente utilizado para estes fins em trabalhos semelhantes, chamado de *probabilidade de cobertura do intervalos de predição* (PCIP), será usado [Veaux et al. 1998], [Hwang and Ding 1997], [Papadopoulos et al. 2001], [Shrestha and Solomatine 2006], [Yang et al. 2002].

O PCIP representa a probabilidade do valor real da variável de interesse y estar localizado dentro do intervalo de predição, e o seu cômputo usualmente é realizado da seguinte forma: para uma determinada rede neural, a porcentagem de valores de y que estão dentro dos respectivos IPs é calculada; quanto mais próxima esta porcentagem estiver do valor nominal para o qual os IPs foram estimados, melhor será a performance do método aplicado para obter os intervalos.

Neste trabalho uma abordagem diferente, apresentada em [Veaux et al. 1998], foi escolhida por se apresentar mais adequada à medição da qualidade dos intervalos estimados [Papadopoulos et al. 2001]. Tal abordagem consiste em se calcular, separadamente, uma PCIP para cada observação y_i de um conjunto de teste. Esta PCIP individual é obtida calculando-se a frequência com que a observação y_i se encontra dentro dos IPs obtidos a partir das RNAs treinadas. Feito isso, uma média das frequências de todas as observações do conjunto de teste é calculada dando origem a uma PCIP para cada conjunto. A PCIP utilizada na análise dos experimentos é então obtida a partir da

média das PCIPs de todos os conjuntos de teste.

Em [Papadopoulos et al. 2001], chama-se a atenção para o fato de, apesar da PCIP servir como medida usual de desempenho de métodos de estimação de intervalos de predição, a mesma reflete apenas a qualidade global dos IPs. É, então, feita a proposta de se calcular o desvio-padrão da distribuição das PCIPs de todas as observações de um conjunto de teste como maneira de avaliar a qualidade dos intervalos localmente. Contudo, chega-se à conclusão de que tal indicador é inapropriado na avaliação do desempenho local de um método de estimação de IPs [Papadopoulos et al. 2001].

Pensando nisto, propõe-se neste trabalho o cálculo de PCIPs específicas para cada vizinhança criada no espaço de entrada. Probabilidades semelhantes entre as vizinhanças indicam PCIPs mais constantes ao longo do domínio da aplicação, e portanto, um melhor desempenho local do método de estimação de IPs.

4.2 Experimento I

Para o primeiro experimento realizado neste trabalho escolheu-se o sistema utilizado na Seção 3.1 como exemplo da deficiência dos IPs obtidos pelo MDC em acusar diferenças na taxa de ruído. O sistema, portanto, é representado pela Equação (3.1), onde ν é um ruído Gaussiano com desvio-padrão dado pela Equação (3.2). Tem-se portanto uma única variável aleatória x representando o vetor de entrada $\mathbf{x}$ e uma variável de interesse y.

As redes usadas neste experimento possuem 5 neurônios na camada oculta e foram treinadas utilizando 50 conjuntos de treinamento contendo 400 elementos e 50 conjuntos com 1000 elementos. Cada elemento de um conjunto de treinamento é formado por um par de observações (x_i, y_i) onde cada observação x_i é gerada aleatoriamente a partir de uma distribuição uniforme dentro do intervalo $[-1, +1]$. A diferença na quantidade de observações se justifica pois desta forma as PCIPs obtidas a partir tanto de redes treinadas com poucos dados de treinamento quanto de redes treinadas com muitos dados serão analisadas, fornecendo uma estimativa de caráter mais geral para o método proposto. No processo de agrupamento dos dados de entrada, um Modelo de Mistura

de Gaussianas (MMG) foi construído utilizando-se o algoritmo *k-médias* para achar os centros de 3 *clusters*. Métodos mais elaborados poderiam ser utilizados neste processo de agrupamento, mas para manter a clareza deste experimento, uma abordagem mais simples foi adotada.

Foram gerados para a etapa de testes 100 conjuntos com 1000 elementos cada, com os valores da variável x sendo amostrados de um Hipercubo Latino, da mesma maneira feita em [Veaux et al. 1998], onde se afirma que tal procedimento possibilita melhor avaliar o desempenho médio dos IPs em termos assintóticos. Os resultados deste experimento podem ser vistos nas Tabelas 4.1 e 4.2.

Tabela 4.1: Experimento I - PCIP para cada método

Método	PCIP$(\%)$
MDC	93,7
MDE	96,3

Tabela 4.2: Experimento I - PCIP para cada vizinhança

Método	Vizinhança 1 PCIP $(\%)$	Vizinhança 2 PCIP $(\%)$	Vizinhança 3 PCIP $(\%)$
MDC	99,9	96,9	84,3
MDE	97,3	96,4	95,1

Como pode ser visto na Tabela 4.1, as PCIPs obtidas por cada um dos métodos são semelhantes em termos absolutos. Enquanto o MDC obtém um desempenho um pouco abaixo do esperado, o MDE alcança uma probabilidade de cobertura um pouco acima do valor nominal de cobertura. Os resultados contidos na Tabela 4.2 contudo, mostram que o MDE apresentou um comportamento mais constante em termos de PCIP do que o MDC, indicando um melhor desempenho local do primeiro. Nela pode-se ver

que as PCIPs do MDC para cada vizinhança apresentam variações consideráveis (principalmente para o *cluster* referente à região mais ruidosa), ao passo que o MDE manteve valores similares de probabilidade de cobertura para todas vizinhanças, demonstrando a robustez do método na presença de ruído variável.

Caso somente os resultados da Tabela 4.1 sejam considerados, seria plausível afirmar que ambos os métodos obtiveram bons desempenhos. Porém, as informações contidas na Tabela 4.2 entram em contradição com tal afirmação. O que ocorre é que, como a estimativa da variância $\hat{\sigma}^2$ utilizada pelo MDC é uma média da variância de todos os resíduos do conjunto de treinamento, a mesma acaba por sobrestimar a variância de alguns padrões de teste (para dados localizados em regiões com pouco ruído) e subestimar a variância de outros (localizados em zonas com muito ruído). Esses erros de estimação acabam por compensar uns aos outros resultando em uma PCIP global próxima ao valor de cobertura desejado mas que mascara o baixo desempenho local do método [Papadopoulos et al. 2001].

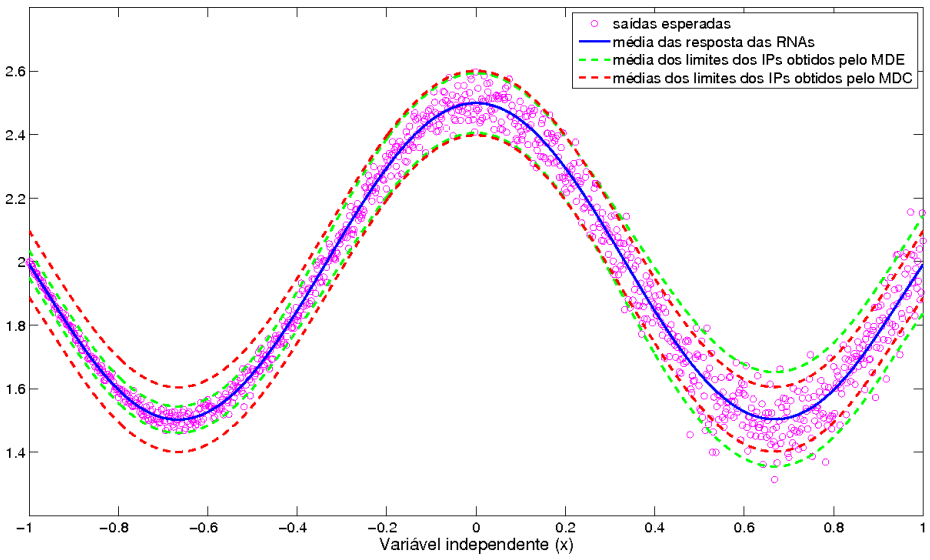

Figura 4.1: Respostas e intervalos de predição referentes a um dos conjuntos de teste. Tanto as respostas como os IPs são valores médios obtidos a partir das estimativas feitas por todas as redes usadas nos experimentos.

Observando a Figura 4.1 fica claro que, para valores de x mais próximos de -1, o MDE fornece IPs mais estreitos que os estimados pelo MDC. À medida que o valor da variável independente se aproxima de $+1$ os intervalos do MDC tendem a ficarem mais largos, refletindo o aumento na taxa de ruído.

A diferença entre as AIPs dos dois métodos fica mais explícita na Figura 4.2. Enquanto as AIPs obtidas pelo MDC se comportam de forma praticamente constante, as AIPs calculadas com o MDE ficam cada vez maiores à medida que o valor da variável x aumenta.

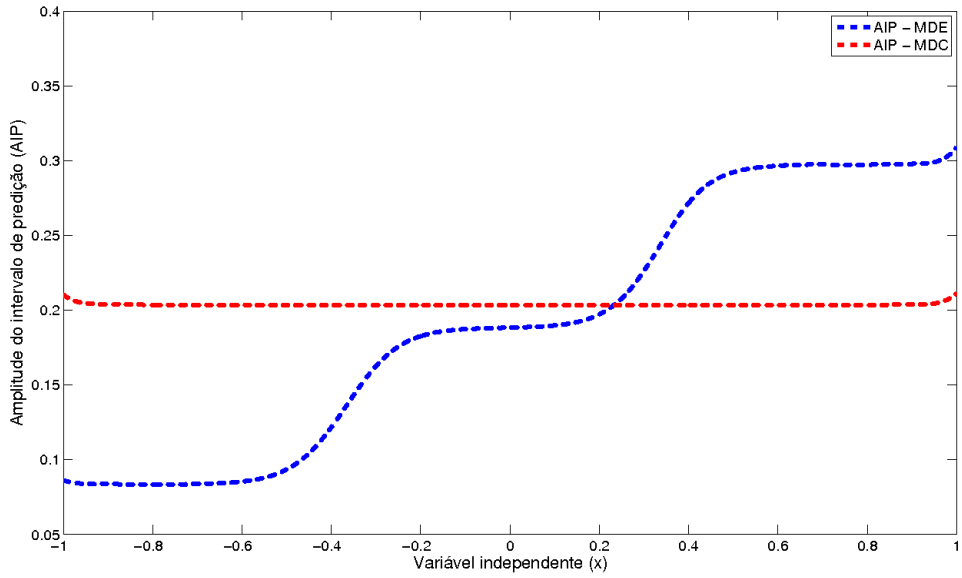

Figura 4.2: Amplitudes médias dos intervalos de predição em função dos dados de entrada de um conjunto de teste.

4.3 Experimento II

O segundo experimento deste trabalho tem como objetivo avaliar os desempenhos do MDC e do MDE na presença de uma mudança mais explícita e abrupta na taxa de ruído. Em vias de se realizar esta análise, a mesma equação do Experimento I é

utilizada como o sistema a ser modelado. A distribuição do ruído adicionado ao sistema, contudo, possui um desvio-padrão σ_ν definido pela Equação (4.1).

$$\sigma_\nu = \left\{ \begin{array}{l} 0,0045 + 0,002x, \text{if } x \leq 0 \\ 0,0045 + 0,01x, \text{if } 0 < x < 0,4 \\ 0,045 + 0,9x, \text{if } x \geq 0,4 \end{array} \right\} \tag{4.1}$$

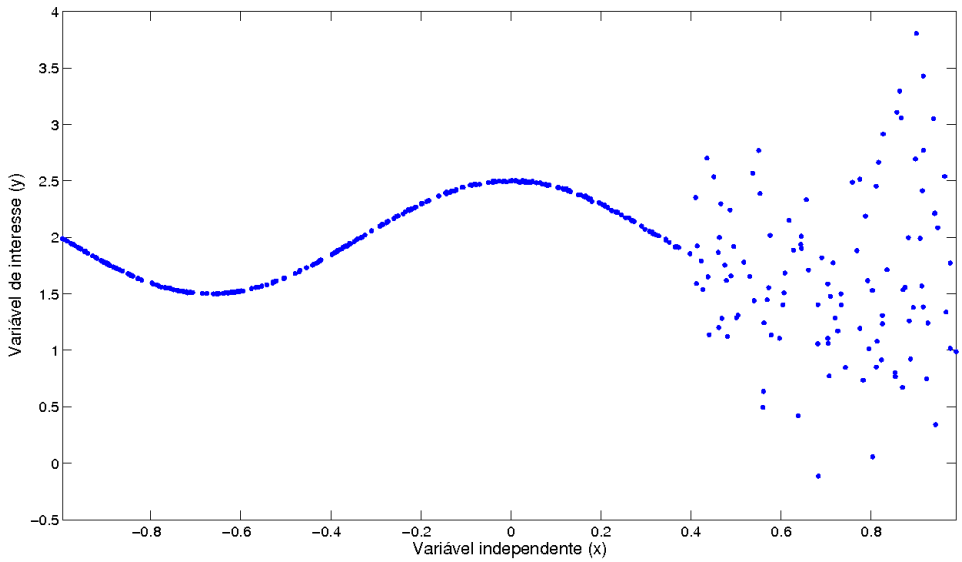

Figura 4.3: Exemplo do comportamento da função a ser modelada no Experimento II.

Com esta modificação, o sistema tratado nesta seção possui um comportamento segundo o qual, somente para valores de x muito próximos a $+1$, tem-se uma alta taxa de ruído (Figura 4.3).

As redes e dados utilizados neste experimento possuem a mesma estrutura do experimento anterior: redes com 5 neurônios na camada escondida, 50 conjuntos de treinamento com 400 observações e 50 conjuntos de 1000 observações. Da mesma forma, os conjuntos de teste foram gerados utilizando o processo aplicado na seção anterior.

Tabela 4.3: Experimento II - PCIP para cada método

Método	PCIP(%)
MDC	92,2
MDE	97,6

Tabela 4.4: Experimento II - PCIP para cada vizinhança

Método	Vizinhança 1 PCIP (%)	Vizinhança 2 PCIP (%)	Vizinhança 3 PCIP (%)
MDC	100	100	76,6
MDE	97,9	99,96	95,1

As Tabelas 4.3 e 4.4 mostram o MDE teve um desempenho semelhante ao MDC, obtendo um valor médio de PCIP um pouco acima do nível de confiança desejado, ao passo que o MDC ficou com uma probabilidade de cobertura abaixo da requerida. Além disso, mais uma vez as PCIPs das vizinhanças obtidas através do método proposto neste trabalho apresentaram uma maior uniformidade do que as obtidas utilizando método tradicional.

Um aspecto interessante, que pode ser observado na Figura 4.4, é que mesmo para valores de x localizados em regiões com baixa taxa de ruído nas observações de y, os IPs estimados a partir do MDC para os valores reais dessas observações são extremamente largos devido à taxa de ruído muito alta em outras regiões do espaço de entrada. Estas AIPs grandes demais podem fazer com que a real precisão das respostas da rede neural para estes dados seja subestimada.

Outro aspecto que deve ser observado é que resultados mais precisos podem ser obtidos mediante a otimização dos parâmetros das curvas gaussianas, o que possibilita a criação de vizinhanças com um nível maior de segregação. Quanto menos

interferência os *clusters* gerados sofrem dos demais, mais realista é a representação do ruído feita pelos IPs.

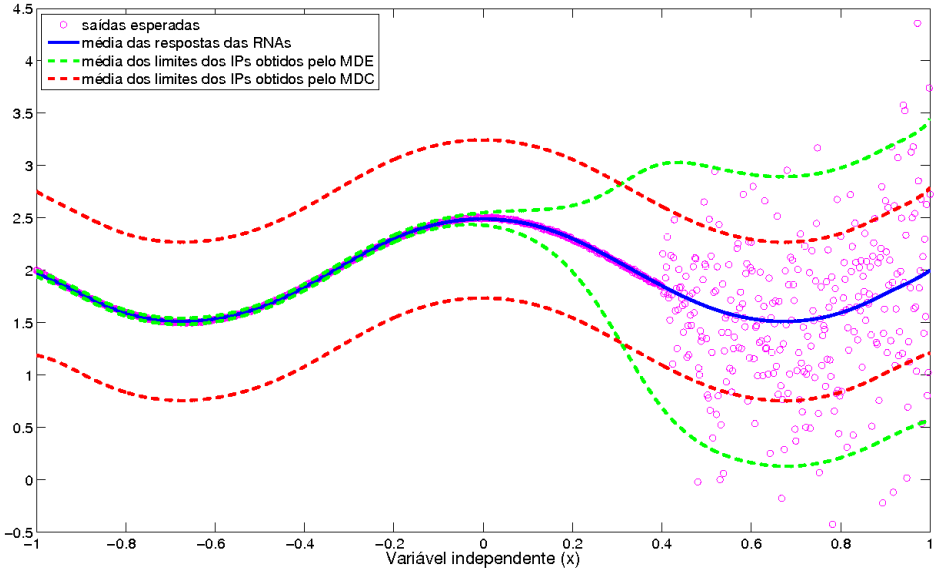

Figura 4.4: Médias das respostas e intervalos de predição referentes a um dos conjuntos de teste.

Este fato pode ser observado na Tabela 4.5, nela são mostrados os resultados obtidos com os mesmos dados utilizados neste experimento, contudo, os parâmetros das gaussianas (centro e abertura) foram otimizados para representar os três intervalos determinados pela Equação 4.1 de forma quase exata. Pode-se ver que as PCIPs das vizinhanças estão bem mais próximas do valor nominal de 95% para o qual os IPs foram estimados.

Tabela 4.5: Experimento II - PCIP para cada vizinhança - parâmetros otimizados

Método	Vizinhança 1 PCIP (%)	Vizinhança 2 PCIP (%)	Vizinhança 3 PCIP (%)
MDC	100	99,5	73,5
MDE	96,5	96,6	95,6

Contudo, essa melhor divisão das vizinhanças acaba por tornar os IPs referentes aos dados de entrada localizados nas fronteiras menos suaves, como pode ser visto nas Figuras 4.5 e 4.6. É necessário, então, que os fatores precisão e suavidade dos IPs sejam balanceados de acordo com as necessidades de cada aplicação em particular.

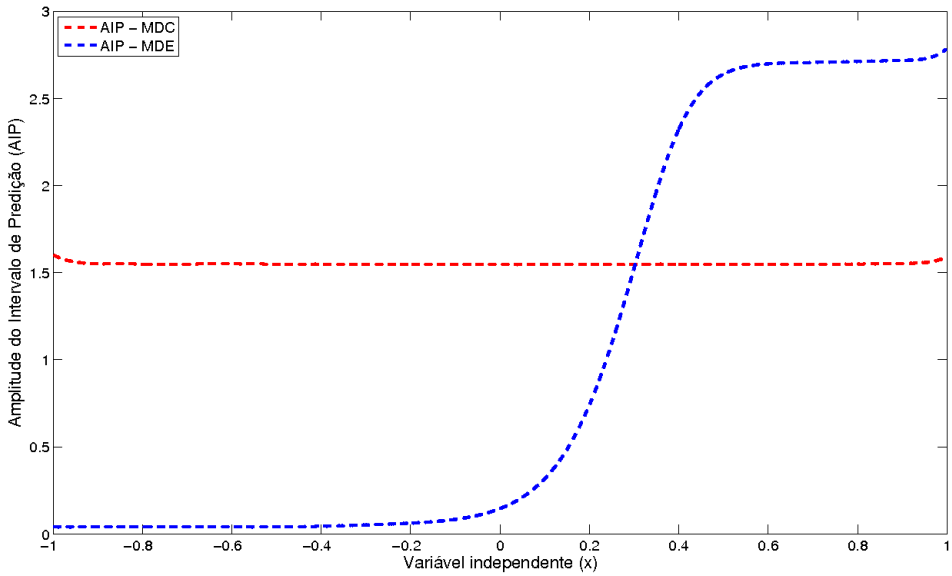

Figura 4.5: AIPs médias em função dos dados de entrada para parâmetros não-otimizados.

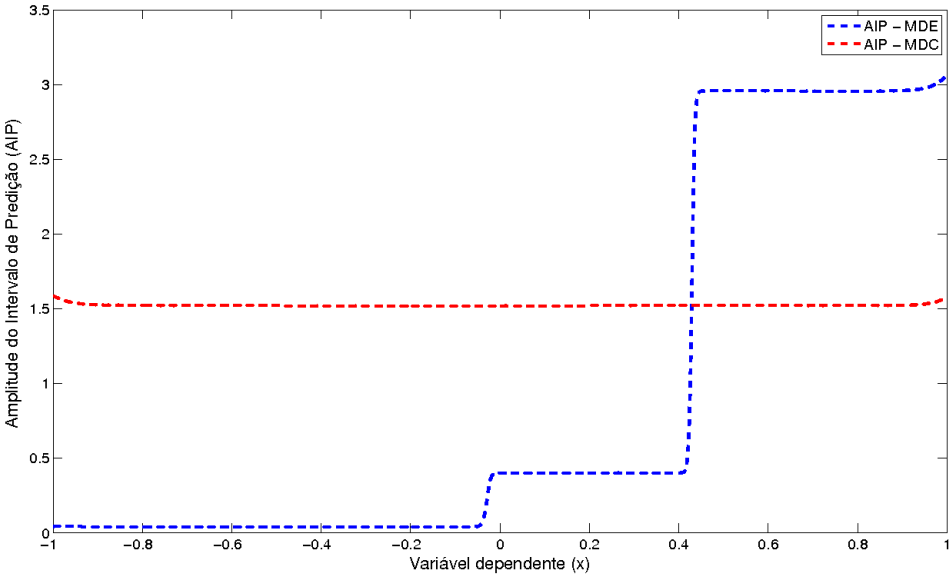

Figura 4.6: AIPs médias em função dos dados de entrada para gaussianas otimizadas.

4.4 Experimento III

O experimento desta seção classifica-se como um experimento sísmico e envolve a aplicação de redes neurais em um problema de caracterização de reservatórios. Neste tipo de aplicação o especialista na área depara-se com a seguinte situação: existem poços perfurados em determinados pontos de uma região com os quais é possível obter valores tanto de atributos sísmicos como de propriedades petrofísicas. O conhecimento destas propriedades é importante pois através delas é possível determinar aspectos cruciais na caracterização de reservatórios. Contudo, os valores de tais propriedades somente são conhecidos em locais onde existem poços perfurados os quais não cobrem toda a região de interesse. As únicas informações disponíveis de toda a região são os atributos obtidos através de um processo de exploração sísmica.

Em trabalhos como [Wong et al. 2002] e [Yang et al. 2002], as redes neurais são aplicadas com o objetivo de modelar uma relação entre os atributos sísmicos e as propriedades petrofísicas, utilizando as informações dos poços perfurados como dados de treinamento. Neste trabalho, o atributo sísmico utilizado como entrada para a rede neural é denominado de variação de impedância de onda compressional ou variação de *impedância-p*, e a propriedade petrofísica com a qual se deseja realizar a caracterização do reservatório é chamada de variação de saturação de água.

A impedância-p diz respeito à resistência que uma determinada rocha apresenta à propagação de uma onda do tipo compressional, ou seja, uma onda que causa um movimento paralelo à direção de seu deslocamento nas partículas do meio no qual se propaga [De Lima 2006]. Como o solo das regiões normalmente é composto por diversos tipos de rochas, tem-se como resultado valores de impedância distintos distribuídos em toda a região de interesse. O atributo sísmico utilizado nos experimentos nada mais é do que a variação destes valores de impedância em um determinado intervalo de tempo.

A saturação de água é uma propriedade petrofísica que indica o volume de água contido nos poros da rocha e a variação desse volume acumulado é que será a

propriedade petrofísica a ser utilizada para caracterizar o reservatório.

Ao conjunto de dados retirados de um determinado poço, quer sejam referentes a atributos sísmicos ou a propriedades petrofísicas, dá-se o nome de *logs* ou perfis. Para este experimento serão utilizados *logs* de 10 poços sintéticos distintos (gerados a partir de um modelo geoestatístico da região) e uma seção sísmica vertical para realizar a caracterização de uma parcela do reservatório.

Uma seção sísmica corresponde a uma fatia retirada de um volume sísmico, o qual contém dados referentes a atributos sísmicos ou propriedades petrofísicas, organizados de maneira a fornecer uma representação espacial da região de interesse. Se este volume contiver informações relacionadas a uma propriedade petrofísica qualquer o valor desta propriedade em uma posição (x, y, z) do volume estará associado à mesma posição na região de interesse.

As seções verticais usadas neste trabalho foram retiradas de forma a obter um conjunto bidimensional de dados no qual o eixo das ordenadas representa a profundidade na região de estudo e o eixo das abcissas está relacionado ao deslocamento em uma das duas direções possíveis na superfície da mesma região, como pode ser visto na Figura 4.7.

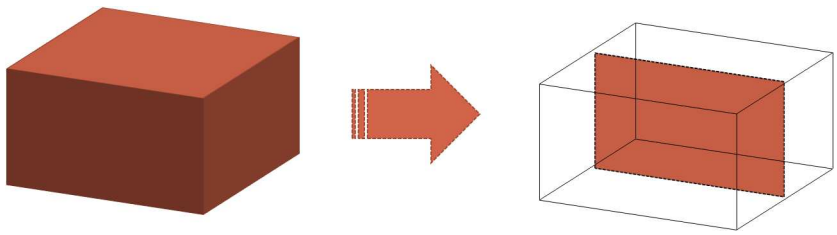

Figura 4.7: Exemplo do formato de uma seção vertical. A partir de um volume sísmico (à esquerda) é retirada a seção sísmica vertical (à direita).

Os valores reais de variação de saturação de água para a seção vertical estudada estão disponíveis (Figura 4.8), possibilitando a avaliação do desempenho dos métodos comparados neste capítulo.

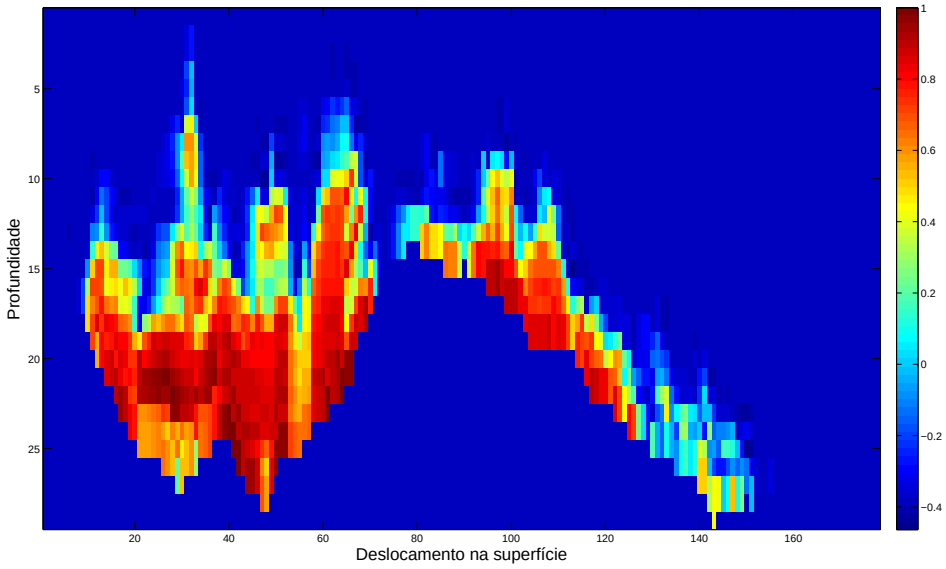

Figura 4.8: Seção sísmica vertical dos valores de variação de saturação de água. A região em destaque representa a área do reservatório compreendida nesta seção.

Foram treinadas 100 redes com a mesma estrutura utilizada nos experimentos hipotéticos utilizando *logs* dos 10 poços. Os dados utilizados no treinamento das redes, os quais foram submetidos a um processo de agrupamento para a geração de 3 vizinhanças no espaço de entrada, podem ser vistos na Figura 4.9.

Observando a Tabela 4.6 pode-se ver que, em termos globais, o MDC obteve um desempenho um pouco melhor que o MDE. Contudo, a Tabela 4.7 revela que o desempenho local do método proposto neste trabalho foi muito superior ao do método tradicional, mantendo PCIPs com um maior grau de similaridade em todas as três vizinhanças do espaço de entrada. Na Figura 4.10 é possível ver que os IPs gerados pelo MDE refletem adequadamente a região mais ruidosa, localizada na vizinhança central do espaço de entrada, apresentando AIPs maiores para as estimativas das redes correspondentes a este local.

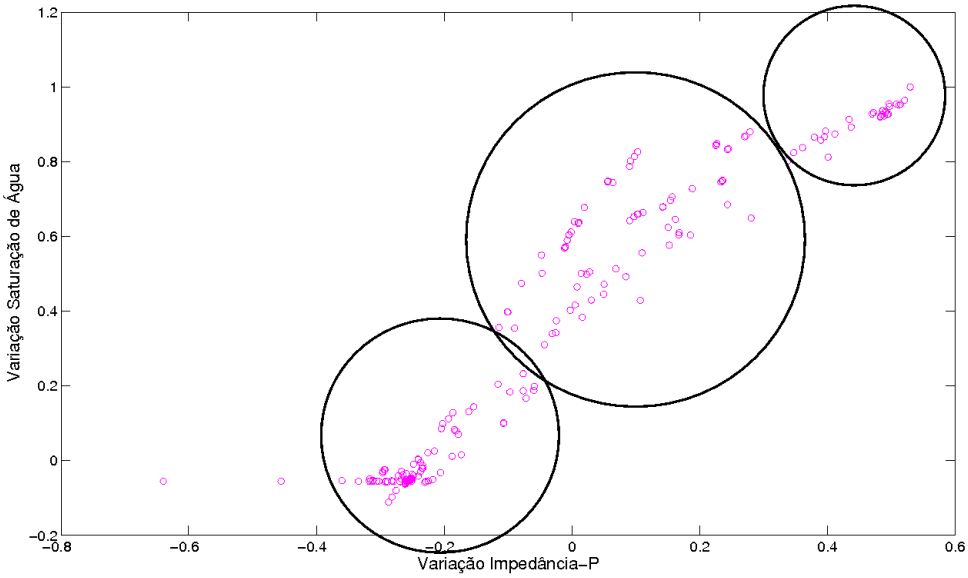

Figura 4.9: Dados de treinamento da rede neural. Os círculos pretos demarcam aproximadamente as regiões onde se acredita que a taxa de ruído seja uniforme.

Tabela 4.6: Experimento III - PCIP para cada método

Método	PCIP(%)
MDC	96,5
MDE	97,3

Tabela 4.7: Experimento III - PCIP para cada vizinhança

Método	Vizinhança 1 PCIP (%)	Vizinhança 2 PCIP (%)	Vizinhança 3 PCIP (%)
MDC	98,9	82,5	95,2
MDE	97,4	98,0	95,6

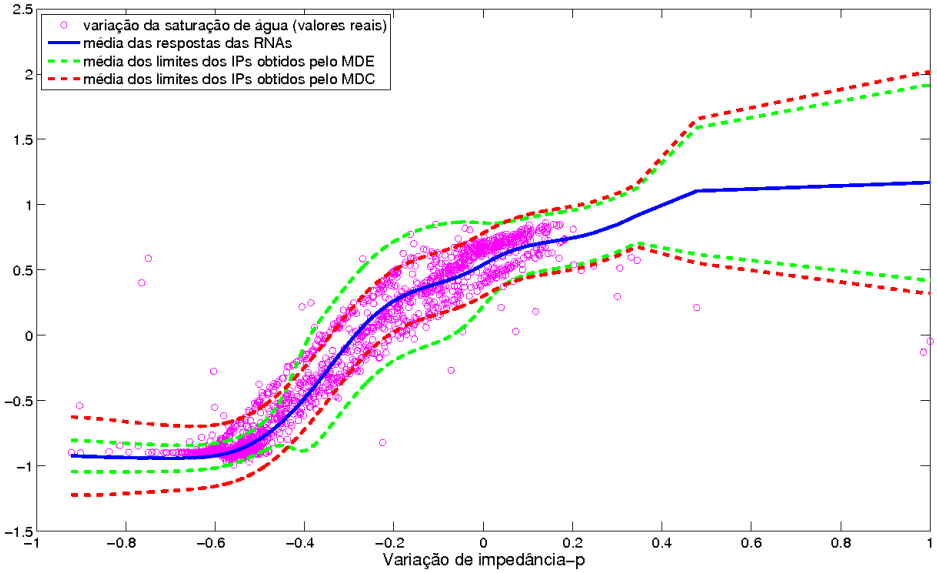

Figura 4.10: Médias das respostas das RNAs e dos IPs estimados para o conjunto de teste.

Na Figura 4.11, tem-se uma representação no mesmo formato da seção sísmica apresentada na Figura 4.8. Porém, ao invés de se utilizar os dados referentes à variação de saturação de água, usou-se os valores de variação de impedância-p do conjunto de teste.

Com o fim de melhor visualizar a segregação, feita pelas vizinhanças, dos dados de entrada, os mesmos foram coloridos de acordo com a faixa de valores correspondentes às vizinhanças. Regiões vermelhas representam dados pertencentes à zona com mais ruído; regiões pintadas de azul se referem aos dados pertencentes à vizinhança localizada à esquerda da vizinhança central vista na Figura 4.9; e as regiões em amarelo representam dados localizados na vizinhança restante. Além destas cores, uma faixa de valores foi pintada de preto. Os dados coloridos desta forma representam regiões de extrapolação, i.e., regiões em que os valores dos dados de entrada estão fora da faixa de valores compreendida pelo conjunto de treinamento, fazendo com que a resposta da rede nesses locais seja pouco confiável.

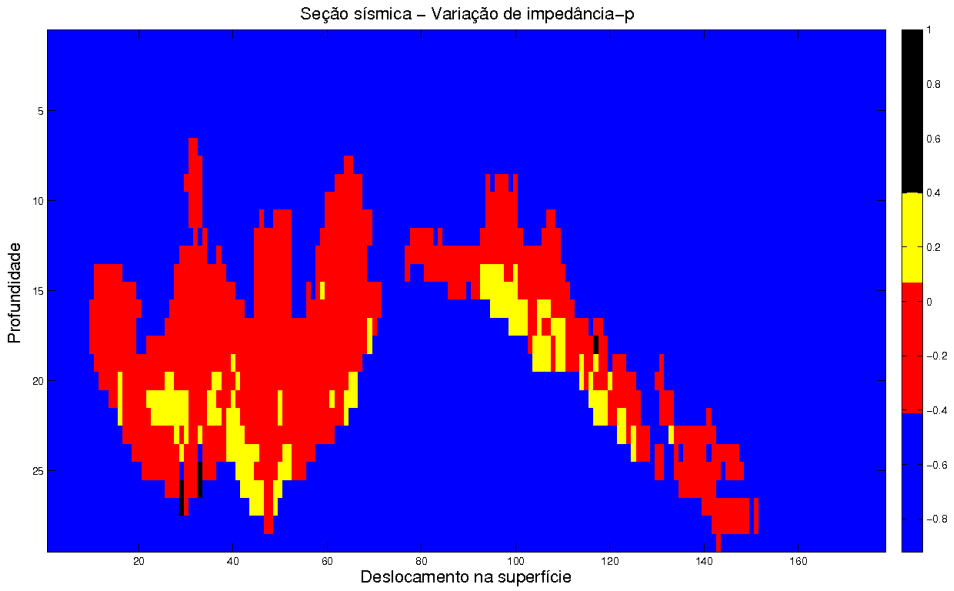

Figura 4.11: Representação da seção sísmica através dos valores da variação de impedância-p.

Através da Figura 4.10 viu-se que os IPs obtidos pelo MDE possuem amplitudes grandes na vizinhança central e vão diminuindo à medida que se aproxima das extremidades do domínio, aumentando somente quando se trata da região de extrapolação (valores acima de 0.37 aproximadamente). Espera-se então que uma representação da seção sísmica, utilizando os valores das AIPs, retrate estas características do reservatório.

Relacionando os IPs estimados pelo MDE com os respectivos dados de entrada, é possível associar as mesmas cores utilizadas na Figura 4.11 para gerar uma representação da seção sísmica utilizando as amplitudes destes intervalos. Assim, AIPs muito grandes, relacionadas aos dados de entrada da região de extrapolação, aparecem em preto; AIPs um pouco menores, correspondentes à vizinhanças mais ruidosa são coloridas de vermelho; e as amplitudes das duas vizinhanças com menos ruído, localizadas à esquerda e direita da vizinhança central na Figura 4.9, aparecem pintadas de azul e amarelo respectivamente.

Esta representação pode ser vista na Figura 4.12 onde, observando a escala de cores é possível ver que as maiores amplitudes (em preto) aparecem nos mesmos locais das regiões de extrapolação da Figura 4.11 (também em preto). As AIPs em verme-

lho (amplitudes grandes) compõem a maior parte da área correspondente ao reservatório, mesma área representada na Figura 4.11 pelos dados pertencentes à vizinhança mais ruidosa. O restante da seção, colorido em azul e amarelo, denota locais cujas amplitudes foram pequenas, de forma muito semelhante ao que ocorre na Figura 4.11 quando se trata dos dados de entrada relacionados às vizinhanças com pouco ruído.

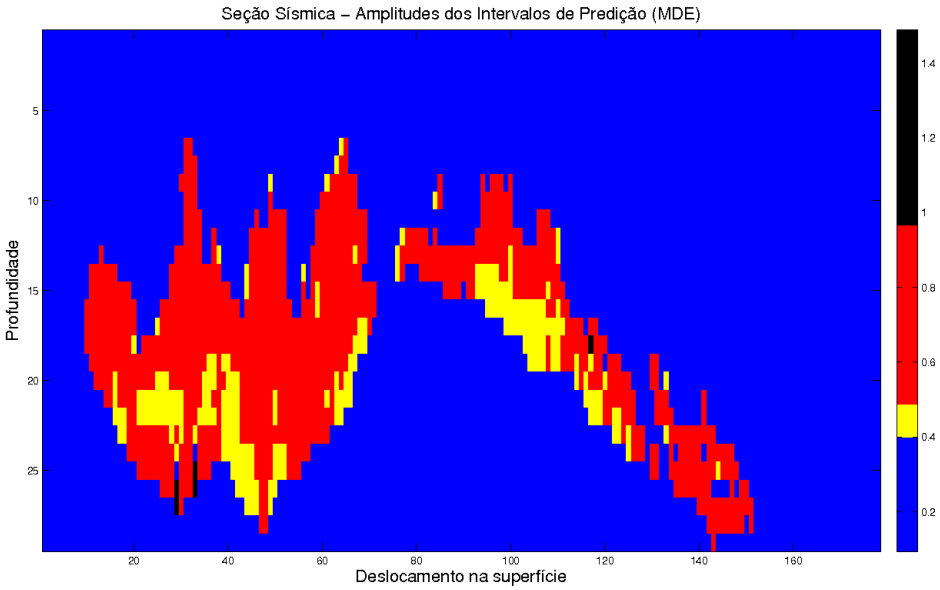

Figura 4.12: Representação da seção sísmica com amplitudes dos intervalos de predição do MDE.

Em suma, as duas figuras são muito semelhantes, demonstrando que através dos intervalos estimados com o MDE, é possível obter uma representação satisfatória da distribuição do ruído na seção sísmica. Esta forma de representação dos valores das AIPs é particularmente interessante quando se lida com um problema onde o vetor de entrada da rede possui muitas dimensões, o que pode ocorrer com frequência em problemas de caracterização de reservatórios, impossibilitando a avaliação dos resultados por meio de gráficos mais simples como o utilizado na Figura 4.10.

Através da observação da Figura 4.12, um especialista da área poderia identificar as regiões em que as predições da rede possuem uma precisão baixa pelo fato dos dados de entrada se encontrarem em uma zona de extrapolação, sendo necessário

portanto que mais *logs* sejam extraídos a fim de cobrir uma área maior do domínio da aplicação. Para as regiões muito ruidosas, as respostas da rede apresentam um nível de incerteza menor, porém continuam com um nível de precisão baixo devido ao ruído, sugerindo uma melhoria nos métodos de aquisição de dados para estas regiões ou na modelagem do sistema.

O mesmo não ocorre com os intervalos obtidos utilizando-se o MDC. As AIPs referentes a este método são praticamente constantes, não permitindo uma representação realista do reservatório por parte dos intervalos, como pode ser visto na Figura 4.13.

Nesta figura foi utilizada a mesma escala de cores da Figura 4.12. Pode-se notar que, apesar do MDC detectar as zonas de extrapolação da rede (em preto), a maior parte dos dados obteve intervalos com amplitudes muito semelhantes, principalmente se for observado que a faixa de cor amarela é relativamente pequena em comparação com as demais faixas de valores.

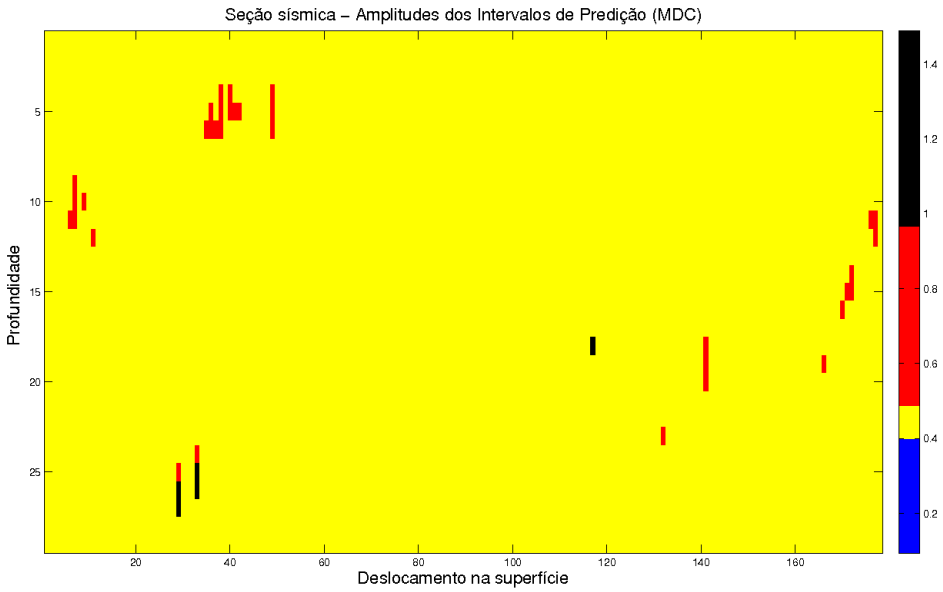

Figura 4.13: Representação da seção sísmica com amplitudes dos intervalos de predição do MDC.

4.5 Experimento IV

Para este último experimento, os parâmetros do Experimento III são mantidos, mudando-se somente o conjunto de dados de treinamento. Desta vez ao invés de poços sintéticos, serão usados *logs* de poços reais perfurados na região estudada.

Ao contrário dos poços sintéticos, gerados a fim de proporcionar uma boa representatividade do sistema para a rede neural, os poços reais não garantem esta propriedade, podendo prejudicar o desempenho do MDE.

As Tabelas 4.8 e 4.9 mostram que nenhum dos dois métodos obteve probabilidades de cobertura adequadas para o nível de confiança desejado (95%). Observando a Figura 4.14 pode-se ver que em média os IPs obtidos possuem amplitudes muito pequenas, mesmo em regiões muito ruidosas, fazendo que boa parte das observações fique fora dos respectivos intervalos.

Tabela 4.8: Experimento IV - PCIP para cada método

Método	PCIP$(\%)$
MDC	83,0
MDE	87,8

Tabela 4.9: Experimento IV - PCIP para cada vizinhança

Método	Vizinhança 1 PCIP $(\%)$	Vizinhança 2 PCIP $(\%)$	Vizinhança 3 PCIP $(\%)$
MDC	96,4	38,7	27,5
MDE	93,6	64,1	66,5

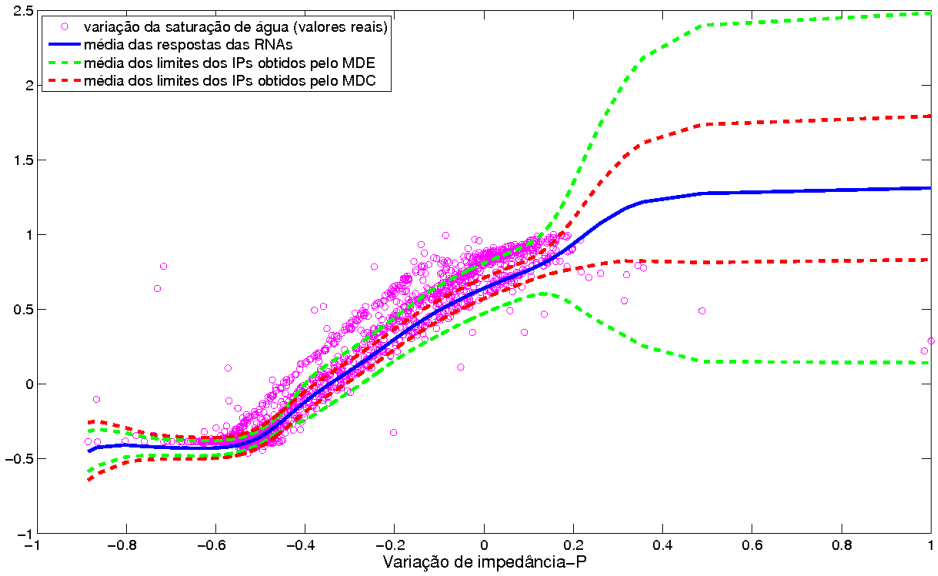

Figura 4.14: Respostas e IPs médios para as redes neurais treinadas com poços reais.

Este desempenho insatisfatório ocorreu pois os dados dos poços reais, utilizados no treinamento das redes, não fornecem informação suficiente sobre o sistema a ser modelado. Observando a Figura 4.15 é possível notar que os dados dos poços reais são menos ruidosos, estando presentes somente em uma faixa de valores dos conjunto de teste. Consequentemente, os resíduos obtidos a partir das redes dizem respeito somente a esta região, fazendo com que a variância estimada pelo MDE seja estimada de forma não-realista e comprometendo o cálculo dos IPs.

O mesmo não ocorre com os poços sintéticos do Experimento III, os quais conseguem retratar de forma adequada o sistema real (Figura 4.16), e assim permitem que os IPs estimados pelo MDE representem a precisão das respostas da RNA de forma coerente. Apesar disso, o MDE apresentou-se como uma escolha mais adequada, obtendo PCIPs nas vizinhanças próximas do valor desejado ou muito maiores que as do MDC.

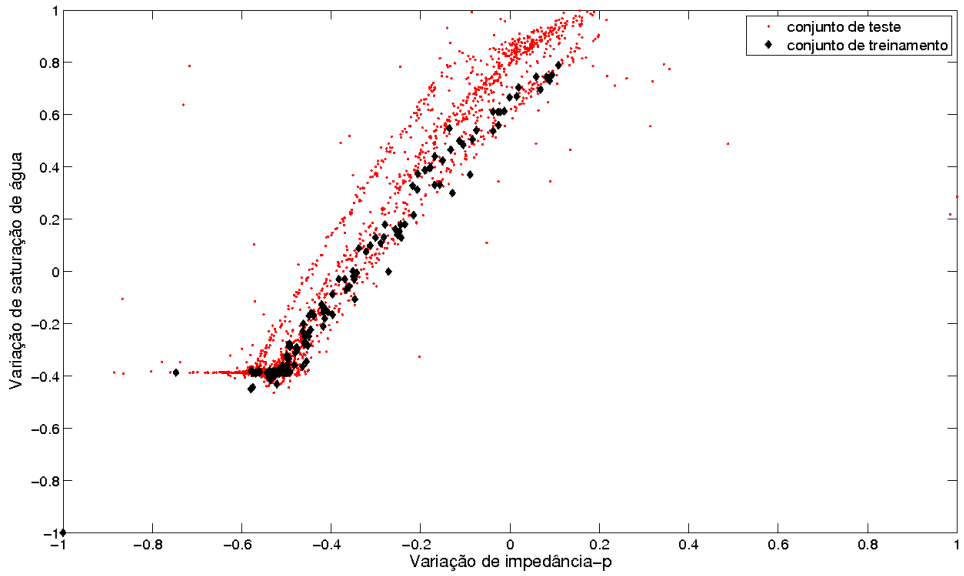

Figura 4.15: Dados utilizados para treinamento e teste das RNAs no Experimento IV.

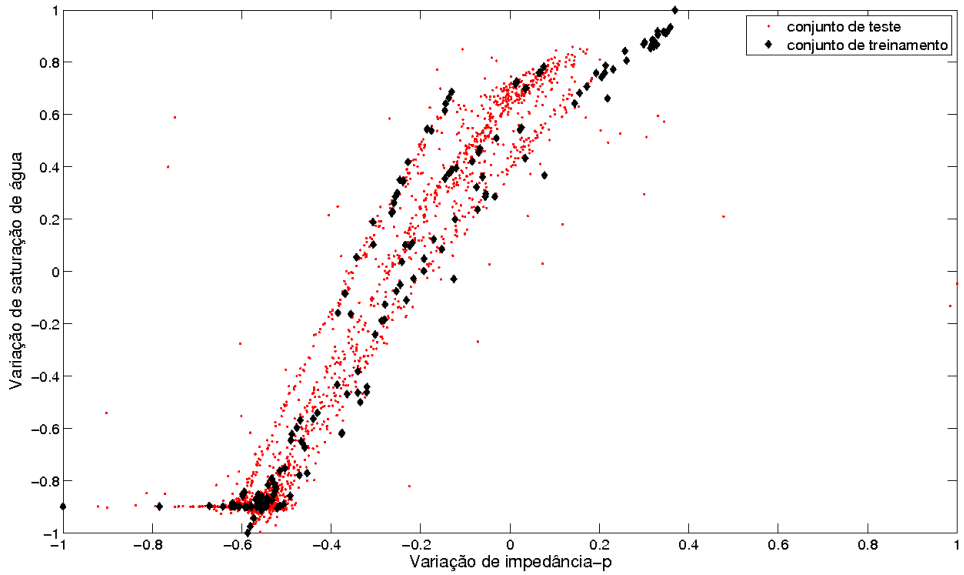

Figura 4.16: Dados utilizados para treinamento e teste das RNAs no Experimento III.

Concluindo, é importante frisar que nos experimentos realizados nas seções 4.2 e 4.3 as características dos sistemas de interesse são bem conhecidas, permitindo

uma modelagem otimizada dos mesmos e reduzindo o erro associado ao modelo de forma que a existência do ruído possa ser atribuída somente aos erros de medição da variável de interesse. Porém, não se pode dizer o mesmo sobre os experimentos sísmicos, para os quais não há garantias de que o sistema foi modelado livre de erros. A predição da variação de saturação de água pode ser prejudicada pela ausência de outros atributos sísmicos que, junto com a variação de impedância-p, revelariam o comportamento do sistema de forma mais precisa. Sendo assim, para estes casos, não é possível afirmar se o ruído causador dos resíduos do modelo provém de erros referentes à medição dos dados ou do erro associado ao modelo.

Capítulo 5

Conclusão

Constatou-se neste trabalho que o cálculo de intervalos de predição para Redes Neurais Artificiais, através do método proposto em [Chryssolouris et al. 1996], não obteve valores de probabilidade de cobertura adequados para problemas em que a taxa de ruído não era uniforme. Nos experimentos realizados foram utilizados dois indicadores, um de caráter global e outro com um enfoque mais local, para medir o desempenho do método delta e da extensão a ele proposta.

Apesar da Probabilidade de Cobertura média dos Intervalos de Predição do Método Delta Clássico obtida nos experimentos estar próxima do valor pretendido, o desempenho local deste método ficou aquém do desejado, com as probabilidades de cobertura variando bastante de vizinhança para vizinhança. O Método Delta Estendido, por outro lado, foi capaz de obter probabilidades de cobertura locais próximas do valor de confiança almejado e ao mesmo tempo com pouca variação entre as vizinhanças. Além disso, também foi possível constatar que as Amplitudes dos Intervalos de Predição do Método Delta Estendido conseguem retratar a distribuição do ruído, sendo mais largas em regiões com maior taxa de ruído e, consequentemente, indicando uma precisão menor para a resposta da rede, ao contrário do que ocorre com o Método Delta Clássico.

5.1 Vantagens e limitações

O uso de um algoritmo de agrupamento e a construção de vizinhanças no espaço de entrada, possibilitam um “afrouxamento” da restrição de uma variância constante dos resíduos, permitindo o cálculo de Intervalos de Predição que retratem de forma realista a precisão das respostas de uma RNA sem exigir modificações no processo de treinamento da rede neural ou em sua estrutura.

Como demonstrado brevemente nos experimentos, melhores resultados podem ser obtidos com o uso de técnicas mais refinadas de agrupamento, as quais consigam particionar o espaço de entrada da aplicação de forma a retratar com mais precisão a distribuição do ruído. Contudo, deve-se tomar um certo cuidado com o nível de segregação gerado pela divisão do espaço de entrada em *clusters*. Caso as vizinhanças estejam muito separadas umas das outras, a suavidade dos Intervalos de Predição estimados será comprometida e os intervalos sofrerão mudanças bruscas nas zonas de transição entre vizinhanças. Da mesma forma, se houver uma sobreposição muito grande entre os *clusters*, a probabilidade de cobertura em cada uma das vizinhanças sofrerá muita interferência do Erro Médio Quadrático das demais, debilitando as suas probabilidades de cobertura.

Outro aspecto importante a ser considerado quando do uso do Método Delta Estendido se refere à quantidade de *clusters* criados no espaço de entrada. Para que a teoria da regressão não-linear possa ser aplicada de forma consistente na estimação de intervalos de predição é necessário que o número de exemplos de treinamento em cada vizinhança seja consideravelmente maior que o número de pesos sinápticos da rede neural. Porém, à medida que se aumenta a quantia de *clusters* no espaço de entrada, menor é o número de observações em cada uma das vizinhanças, o que pode levar ao cálculo Intervalos de Predição que não representem corretamente a precisão da resposta da RNA ou até mesmo à impossibilidade de se obter tais intervalos. Tais aspectos não foram alvos de estudos mais aprofundados por fugirem do intento deste trabalho, o qual visa demonstrar os princípios que norteiam o método proposto nesta dissertação e o ganho em desempenho obtido pela aplicação dos mesmos.

Como dito na Seção 3.1, o ruído presente em um sistema pode ser cau-

sado tanto por erros de medição das observações da variável de interesse como por erros associados ao modelo, e não foi parte dos objetivos deste trabalho estimar separadamente a influência dos mesmos nos Intervalos de Predição, como feito em [Nix and Weigend 1995], correndo-se o risco de aumentar demasiadamente o custo computacional do método proposto nesta dissertação. Desta forma não é possível afirmar qual das duas causas é efetivamente responsáveis por amplitudes de intervalos muito largas.

Por fim, em todas as situações abordadas nos experimentos foi considerado que os dados de treinamento estão distribuídos uniformemente, e portanto a questão relativa à densidade de dados no domínio da aplicação, abordadas em [Leonard et al. 1992] e [Shao et al. 1997], não foi tratada nesta dissertação, uma vez que o método aqui proposto tem como objetivo contornar somente a restrição referente à variância dos resíduos.

5.2 Sugestões para trabalhos futuros

Os resultados obtidos pelo Método Delta Estendido nesta dissertação demonstram que o desempenho do mesmo supera o do Método Delta Clássico. Contudo, com o fim de obter informações que corroborem para o uso da abordagem aqui proposta, sugere-se que experimentos com dados de entrada multidimensionais sejam feitos, uma vez que os experimentos realizados no Capítulo 4 utilizaram somente dados unidimensionais, com o intuito de tornar a visualização do ruído, e a posterior análise dos resultados obtidos, mais simples.

Os métodos propostos em [Leonard et al. 1992] e [Nix and Weigend 1995] também buscam contornar a restrição relacionada à variância dos resíduos, contudo em nenhum dos dois trabalhos é utilizado qualquer tipo de indicador de performance local para avaliar o desempenho dos mesmos. Experimentos comparativos entres estes métodos e o Método Delta Estendido, se realizados, permitiriam a escolha do método mais adequada em aplicações reais. Tais experimentos não foram executados no presente trabalho, uma vez que buscou-se demonstrar primeiramente que o método proposto obteria resultados no mínimo equivalentes ao método tradicional já utilizado.

Um aspecto interessante a ser estudado em trabalhos futuros trata da incorporação da influência da densidade dos dados de treinamento no cálculo dos intervalos de predição, fazendo com que estes intervalos consigam retratar, além da distribuição do ruído, a incerteza nas respostas da rede neural devido à baixa quantidade de informação disponível em uma região do espaço de entrada. A adição desta capacidade ao Método Delta Estendido possibilitaria, portanto, que o mesmo fosse utilizado em um número maior de aplicações reais.

Outro aspecto que merece ser alvo de estudos diz respeito à separação entre os erros de medição das observações da variável de interesse e os erros associados ao modelo quando do cálculo dos Intervalos de Predição, o que possibilitaria a identificação da principal fonte geradora de incertezas nas respostas da rede neural para um sistema em específico.

Por fim, com o intuito de amenizar os efeitos da limitação do Método Delta Estendido referente à quantidade de *clusters* nos quais o espaço de entrada pode ser particionado, uma investigação sobre a viabilidade de se estimar os pesos sinápticos mais atuantes, em função de um vetor de entrada específico, possibilitaria que um número maior de vizinhanças pudessem ser criadas no espaço de entrada da rede neural.

Referências Bibliográficas

[Bishop 1995] Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, New York, NY.

[Boggs 2009] Boggs, R. (2009). Exploring data website. Disponível em: http://exploringdata.cqu.edu.au/. Propriedade da Education Queensland.

[Chinman and Ding 1998] Chinman, R. B. and Ding, J. (1998). Prediction limit estimation for neural network models. *IEEE Transactions on Neural Networks*, 9(6):1515–1522.

[Chryssolouris et al. 1996] Chryssolouris, G., Lee, M., and Ramsey, A. (1996). Confidence interval prediction for neural network models. *IEEE Transactions on Neural Networks*, 7(1):229–232.

[De Lima 2006] De Lima, K. T. P. (2006). *Utilização de Métodos Sísmicos, Perfilagem e Testemunhos de Poços Para Caracterização dos Turbiditos da Formação Urucutuca na Bacia de Almada (BA)*. PhD thesis, Universidade Estadual do Norte Fluminense.

[Dybowski and Roberts 2001] Dybowski, R. and Roberts, S. J. (2001). Confidence intervals and prediction intervals for feed-forward neural networks. In *Clinical Applications of Artificial Neural Networks*, pages 298–326. University Press.

[Haykin 1998] Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall, second edition.

[Heskes 1997] Heskes, T. (1997). Practical confidence and prediction intervals. In *Advances in Neural Information Processing Systems*, volume 9, pages 176–182. MIT press.

[Hwang and Ding 1997] Hwang, G. J. T. and Ding, A. A. (1997). Prediction intervals for artificial neural networks. *Journal of the American Statistical Association*, 92(438):748–757.

[Lebaron and Weigend 1994] Lebaron, B. and Weigend, A. S. (1994). Evaluating neural network predictors by bootstrapping. In *Computer Science Department, University of Colorado at Boulder, ftp://ftp.cs.colorado.edu/pub/TimeSeries/MyPapers/bootstrap.ps*, pages 1207–1212.

[Leonard et al. 1992] Leonard, A., Kramer, M., and Ungar, L. (1992). A neural network architecture that computes its own reliability. *Computers and Chemical Engineering*, 16(9):819–835.

[Nix and Weigend 1995] Nix, D. and Weigend, A. (1995). Learning local error bars for nonlinear regression. In Tesauro, G., Touretzky, D. S., and Leen, T. K., editors, *Advances in Neural Information Processing Systems*, volume 7. MIT Press, Cambridge, MA.

[Papadopoulos et al. 2001] Papadopoulos, G., Edwards, P., and Murray, A. (2001). Confidence estimation methods for neural networks: a practical comparison. *IEEE Transactions on Neural Networks*, 12(6):1278–1287.

[Seber and Wild 2003] Seber, G. and Wild, C. (2003). *Nonlinear regression*. Wiley-Interscience, New York, NY.

[Shao et al. 1997] Shao, R., Martin, E., Zhang, J., and Morris, A. (1997). Confidence bounds for neural network representations. *Computers and Chemical Engineering*, 21:1173–1178(6).

[Shrestha and Solomatine 2006] Shrestha, D. L. and Solomatine, D. P. (2006). Machine learning approaches for estimation of prediction interval for the model output. *Neural Networks*, 19(2):225–235.

[Veaux et al. 1998] Veaux, R. D., Schweinsberg, J., Schumi, J., and Ungar, L. (1998). Prediction intervals for neural networks via nonlinear regression. *Technometrics*, 40(4):273–282.

[Wong et al. 2002] Wong, P. M., Bruce, A. G., and Gedeon, T. D. (2002). Confidence bounds of petrophysical predictions from conventional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 40:1440–1444.

[Yang et al. 2002] Yang, L., Kavli, T., Carlin, M., Clausen, S., and Groot, P. F. M. D. (2002). An evaluation of confidence bound estimation methods for neural networks. In *Advances in Computational Intelligence and Learning: Methods and Applications*, pages 71–84, Deventer, The Netherlands, The Netherlands. Kluwer, B.V.

Apêndice A

Algoritmos de Otimização de Parâmetros

A.1 Conceitos Gerais

A.2 Gradiente Descendente

O algoritmo chamado de Gradiente Descendente, também conhecido como *steepest descent*, é um dos métodos mais simples usados no treinamento de RNAs. Neste algoritmo o valor de atualização dos pesos da RNA é calculado iterativamente, com a atualização do vetor de pesos $w^{(\tau)}$ para a iteração τ dada por:

$$\Delta \mathbf{w}^{(\tau)} = -\eta \nabla E^n|_{\mathbf{w}^{(\tau)}} \tag{A.1}$$

Onde:

- n indica o padrão de treinamento apresentado à RNA;
- η representa taxa de aprendizado, parâmetro responsável por determinar quão rápido os pesos da rede serão modificados a cada iteração;
- $\nabla E^n|_{\mathbf{w}^{(\tau)}}$ é o gradiente do erro (E) para o padrão n em relação ao vetor de pesos $\mathbf{w}$ no instante τ.

Desta forma, cada vez que o vetor de pesos é atualizado, ele irá se mover na direção em que haja maior diminuição do erro E. Este processo pode ser visualizado na figura abaixo:

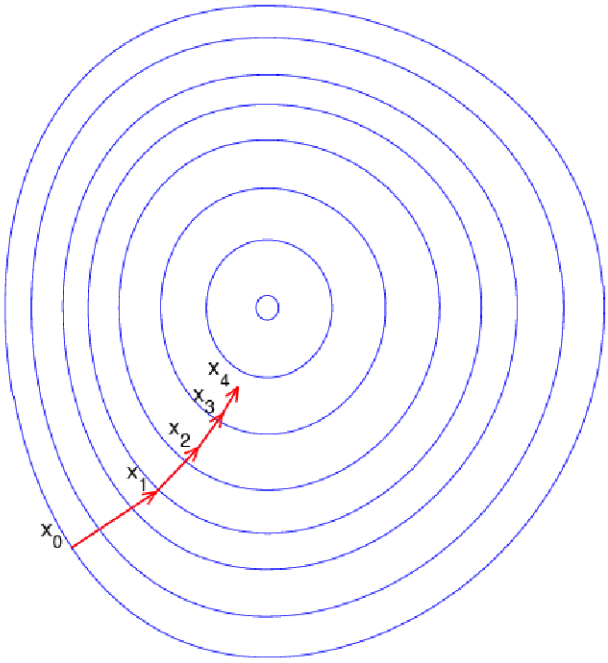

Figura A.1: Exemplo do comportamento do algoritmo Gradiente Descendente. As linhas azuis são curvaturas do erro e quanto mais externa a linha maior é o valor deste.

O Gradiente Descendente, apesar de ser um algoritmo relativamente fácil de ser implementado, apresenta algumas desvantagens. A primeira diz respeito à escolha da taxa de aprendizado, a qual deve ser escolhida com cuidado uma vez que um valor muito alto fará com que os pesos oscilem demasiadamente, resultando num possível aumento de E e impedindo que o algoritmo convirja para um mínimo; caso o valor de η seja muito baixo o aprendizado da rede ocorrerá muito lentamente. A outra desvantagem deste algoritmo decorre do fato do gradiente, usado como base para direcionar a atualização dos pesos, ser local e não apontar diretamente para o mínimo da função de erro. Em um problema em que que a curvatura do erro varie muito com a direção adotada, o Gradiente Descendente se comporta de maneira ineficiente, tendo que tomar vários passos pequenos para alcançar o mínimo (Figura A.2) [Bishop 1995].

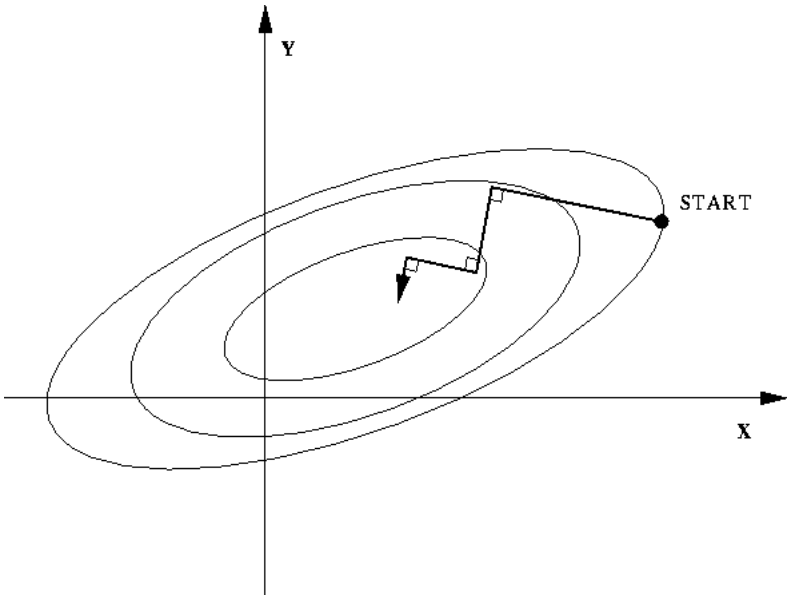

Figura A.2: Exemplo do comportamento em "zigue-zague"do Gradiente Descendente.

A.3 Método de Newton

O Método de Newton faz parte de uma classe de algoritmos que, ao contrário do Gradiente Descendente, utilizam a matriz Hessiana explicitamente. Através de uma expansão de Taylor, uma aproximação quadrática local da função de erro pode ser escrita como:

$$E(\mathbf{w}^{(\tau)} + \Delta\mathbf{w}) = E(\mathbf{w}^{(\tau)}) + \Delta\mathbf{w}^{\mathbf{T}}\nabla E(\mathbf{w}^{(\tau)}) + \frac{1}{2}\Delta\mathbf{w}^{\mathbf{T}}\mathbf{H}\Delta\mathbf{w} \qquad \text{(A.2)}$$

A partir da Equação (A.2) o gradiente desta aproximação, avaliado no vetor $\Delta\mathbf{w}$, é dado por:

$$\mathbf{g} = \nabla E(\mathbf{w}^{(\tau)} + \Delta\mathbf{w}) = \nabla E(\mathbf{w}^{(\tau)}) + \mathbf{H}\Delta\mathbf{w} \qquad \text{(A.3)}$$

Dito isto, consideremos que o vetor de pesos $\mathbf{w}^{(\tau)}$ esteja a uma distância $\Delta\mathbf{w}$ próxima o suficiente do vetor $\mathbf{w}^*$, o qual minimiza a função de erro. Sabendo que o gradiente de tal função em $\mathbf{w}^*$ é dado por $\nabla E(\mathbf{w}^*) = 0$, infere-se que $\mathbf{H}\Delta\mathbf{w} = -\nabla E(\mathbf{w}^{(\tau)})$ e portanto $\mathbf{w}*$ pode ser escrito como:

$$\mathbf{w}^* = \mathbf{w} - \mathbf{H}^{-1}\mathbf{g} \tag{A.4}$$

O termo $\mathbf{H}$ que aparece na Equação A.3 é chamado de matriz Hessiana e cada elemento desta é representado por:

$$\mathbf{H}_{ij} \equiv \frac{\partial^2 E}{\partial w_i \partial w_j} \tag{A.5}$$

O vetor $-\mathbf{H}^{-1}\mathbf{g}$, conhecido como direção de Newton ou passo de Newton é o responsável por indicar a direção em que os pesos devem ser atualizados e, diferente do gradiente usado no Gradiente Descendente, aponta diretamente para o ponto mínimo da função de erro independente do vetor $\mathbf{w}$ sobre o qual ele é avaliado. Desta forma o método de Newton consegue convergir rapidamente e de modo assintótico sem apresentar o comportamento de ziguezague que algumas vezes é observado no comportamento do método Gradiente Descendente, como mostrado na Figura A.3.

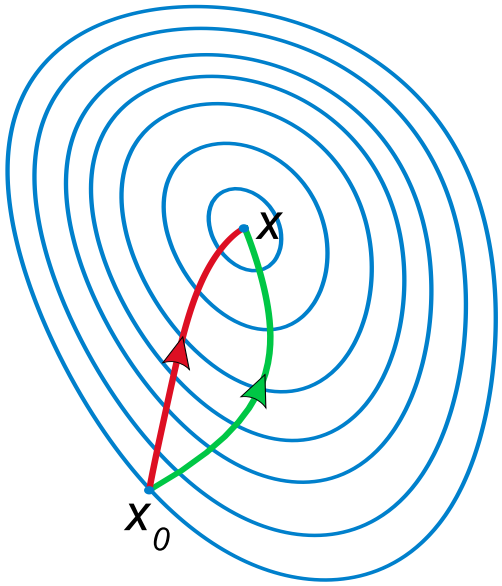

Figura A.3: Exemplo do comportamento do Método de Newton. A linha verde denota o caminho obtido com o Gradiente Descendente enquanto que a linha vermelha representa o a direção obtida com o Método de Newton

Apesar do Método de Newton apresentar uma performance superior na tarefa de encontrar o mínimo da função de erro, sua aplicação acaba se mostrando muito custosa na prática. A aproximação obtida através da Equação (A.2), por ser uma aproximação, não é exata e este fato torna necessário o cálculo de tal aproximação repetidas vezes e de forma iterativa, com a matriz Hessiana sendo recalculada a cada iteração bem como a sua inversa. Segundo [Bishop 1995] o cálculo de $\mathbf{H}$ possui uma complexidade $O(NW^2)$ enquanto que são necessários $O(W^3)$ passos para calcular sua inversa, tornando o custo computacional deste método muito alto.

A.4 Levenberg-Marquardt

Diferentemente dos dois métodos vistos anteriormente, os quais minimizam uma grande variedade de funções, o algoritmo Levenberg-Marquardt (LM), foi especificamente concebido para minimizar uma soma de erros quadráticos na forma:

$$E = \frac{1}{2}\sum_{i}(\epsilon_i)^2 \tag{A.6}$$

Assim como o método de Newton, este algoritmo também faz uso da matriz Hessiana para achar um mínimo para a função de erro. Contudo, ao invés de calculá-la diretamente, uma aproximação da mesma é utilizada:

$$\mathbf{H} = \mathbf{Z^T Z}. \tag{A.7}$$

onde $\mathbf{Z}$ representa a matriz Jacobiana, constituída pelas derivadas de primeira ordem da função de erro em relação aos pesos da rede:

$$\mathbf{Z}_{ij} \equiv \frac{\partial \epsilon_i}{\partial w_j} \tag{A.8}$$

Sendo assim, o custo computacional exigido para o cálculo da matriz Hessiana é reduzido já que as derivadas parciais de primeira ordem em relação aos pesos da RNA são facilmente obtidos durante a execução do algoritmo de *backpropagation*, restando somente o custo de cálculo de $\mathbf{H}^{-}1$.

Um problema inerente ao uso da matriz Hessiana se deve ao fato desta ter de ser necessariamente definida positivamente, ou seja, ela deve possuir uma matriz inversa. O algoritmo LM contorna este problema ao adicionar à aproximação da matriz Hessiana uma matriz definida positivamente que consiste da matriz identidade $\mathbf{I}$ de $\mathbf{H}$ multiplicada por um fator λ suficientemente grande.

Sob o ponto de vista geométrico, este procedimento pode ser visto como um "encurtamento" do passo dado pelo algoritmo. Quando a linearização da função de erro é feita usando-se uma série de Taylor até o segundo termo, o resultado obtido é uma aproximação da função de interesse por uma parábola. A partir desta aproximação tenta-se chegar ao ponto de mínimo da parábola. Caso o passo do algoritmo seja muito grande, pode ocorrer do mesmo acabar passando do ponto de mínimo resultando na não-convergência do algoritmo. O "encurtamento" do passo age como um meio de prevenir que isto ocorra, garantindo assim a convergência do LM. Pelo fato do modelo usado pelo algoritmo ser confiável somente dentro de uma região em torno do ponto de busca, o Levenberg-Marquardt é considerado um exemplo de método de região de confiança (*model trust region*).

Por fim ao minimizarmos a função de erro em relação aos pesos da iteração $\tau + 1$ temos:

$$\mathbf{w}^{\tau+1} = \mathbf{w}^{\tau} - (\mathbf{Z}^{\mathbf{T}}\mathbf{Z} + \lambda\mathbf{I})^{-1}\mathbf{Z}^{\mathbf{T}}\epsilon(\mathbf{w}^{\tau}) \tag{A.9}$$